Augmented Reality Assistance for Surgical Interventions using Optical See-Through Head-Mounted Displays

by

Long Qian

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

April, 2020

Abstract

Augmented Reality (AR) offers an interactive user experience via enhancing the real world environment with computer-generated visual cues and other perceptual information. It has been applied to different applications, e.g. manufacturing, entertainment and healthcare, through different AR media. An Optical See-Through Head-Mounted Display (OST-HMD) is a specialized hardware for AR, where the computer-generated graphics can be overlaid directly onto the user's normal vision via optical combiners. Using OST-HMD for surgical intervention has many potential perceptual advantages. As a novel concept, many technical and clinical challenges exist for OST-HMD-based AR to be clinically useful, which motivates the work presented in this thesis.

From the technical aspects, we first investigate the display calibration of OST-HMD, which is an indispensable procedure to create accurate AR overlay. We propose various methods to reduce the user-related error, improve robustness of the calibration, and remodel the calibration as a 3D-3D registration problem. Secondly, we devise methods and develop hardware prototype to increase the user's visual acuity

of both real and virtual content through OST-HMD, to aid them in tasks that require high visual acuity, e.g. dental procedures. Thirdly, we investigate the occlusion caused by the OST-HMD hardware, which limits the user's peripheral vision. We propose to use alternative indicators to remind the user of unattended environment motion.

From the clinical perspective, we identified many clinical use cases where OST-HMD-based AR is potentially helpful, developed applications integrated with current clinical systems, and conducted proof-of-concept evaluations. We first present a "virtual monitor" for image-guided surgery. It can replace real radiology monitors in the operating room with easier user control and more flexibility in positioning. We evaluated the "virtual monitor" for simulated percutaneous spine procedures. Secondly, we developed *ARssist*, an application for the bedside assistant in robotic surgery. The assistant can see the robotic instruments and endoscope within the patient body with *ARssist*. We evaluated the efficiency, safety and ergonomics of the assistant during two typical tasks: instrument insertion and manipulation. The performance for inexperienced users is significantly improved with *ARssist*, and for experienced users, the system significantly enhanced their confidence level. Lastly, we developed *ARAMIS*, which utilizes real-time 3D reconstruction and visualization to aid the laparoscopic surgeon. It demonstrates the concept of "X-ray see-through" surgery. Our preliminary evaluation validated the application via a peg transfer task, and also showed significant improvement in hand-eye coordination.

ABSTRACT

Overall, we have demonstrated that OST-HMD based AR application provides ergonomic improvements, e.g. hand-eye coordination. In challenging situations or for novice users, the improvements in ergonomic factors lead to improvement in task performance. With continuous effort as a community, optical see-through augmented reality technology will be a useful interventional aid in the near future.

Thesis Committee

Peter Kazanzides, Research Professor, Johns Hopkins University, Baltimore, USA

Nassir Navab, Professor, Technical University of Munich, Munich, Germany

Russell H. Taylor, Professor, Johns Hopkins University, Baltimore, USA

Simon P. DiMaio, Director, Intuitive Surgical Inc., Sunnyvale, USA

Acknowledgments

First and foremost, I would like to thank my primary advisor, Prof. Peter Kazanzides. He offered me the opportunity to work as a summer intern in the Laboratory for Computational Sensing and Robotics (LCSR) in 2014. The research experience greatly motivated me to join LCSR as a PhD student. Prof. Kazanzides has been supporting my research and helping me mature as a person throughout the five years of life at Johns Hopkins University. He supported my interest in augmented reality at the beginning of my PhD study. He also supported my decision to experience the industry with 6-month internship in California in 2017. Moreover, I often came up with research ideas that do not fully overlap with my funding project, but Prof. Kazanzides would not hesitate to grant me the freedom of exploring my idea, even with his own discretionary funding. Prof. Kazanzides is also a role model for me. His rationality, responsibility, kindness and patience will continue to guide me in my life. Without his support and guidance, the materials in this dissertation will not be possible.

I would like to thank Prof. Nassir Navab for his supervision for my research.

ACKNOWLEDGMENTS

Prof. Navab is always available to chat about research ideas and many other exciting things. His enthusiasm and vision in research has greatly inspired me. He offered me many opportunities to attend conferences, workshops and build connections with fellow researchers, which helped me to become a professional researcher.

I would like to thank Prof. Russell Taylor and Dr. Simon P.DiMaio for serving on my thesis committee. Prof. Russell Taylor is a true pioneer in computer-integrated surgery. His lecture of CIS introduced many generations of researchers, including me, to this exciting field of research. Dr. Simon P.DiMaio offered me the opportunity to experience the life in the industry, and gave me many advice about research and life.

I would like to thank Prof. Louis L. Whitcomb, Prof. Simon Leonard and Prof. Iulian Iordachita for participating in my GBO committee, and sharing me with their comments and knowledge. I would like to thank Prof. Joanne Selinski and Prof. Noah Cowan for offering the opportunity to assist them in teaching. I would like to thank Anton Deguet for his support on the da Vinci Research Kit, his help on Linux programming and debugging, and sharing various administrative passwords with me. I would like to thank Prof. Yunhui Liu and Dr. Zerui Wang for offering the opportunity to work as a visiting scholar at Chinese University of Hong Kong.

I would like to thank my many collaborators helping me to turn ideas into reality, including Dr. Zihan Chen, Prof. Mathias Unberath, Javad Fotouhi, Jie Ying Wu, Ehsan Azimi, Alexander Winkler, Xiran Zhang, Dr. Alexander Plopski, Dr. Bernhard Fuerst, Alexander Barthel, Dr. Alex Johnson, Dr. Greg Osgood, Kevin Yu, Jianren

ACKNOWLEDGMENTS

Wang and Dr. Rafa Rahman. I would like to thank my friends that make this journey remarkable, including Jiayan Gu, Xianjun Zhang, Singchun Lee, Zhaoshuo Li, Cong Gao, Chenxi Liu, Huiyu Wang, Lichen Fang, Daniil Pakhomov, Zerui Wang, Chengzhi Song, Qi Luo, Jiahao Wu, Yiwei Jiang, Andrea Mariani, Edoardo Pellegrini, Tommaso Da Col, Chloé Audigier, Nikita Ivkin, Alexis Cheng, Dafang Wang. I would also like to thank Dr. Omid Mohareri, Govinda Payyavula, Cortney Jansen and Dale Bergman from Intuitive Surgical Inc. for supporting my research.

Finally, I would like to thank my parents, Yonggen Qian and Yajuan Li for their constant love, support and encouragement.

Dedication

This thesis is dedicated to my parents.

Contents

List of Tables

LIST OF TABLES

List of Figures

Chapter 1

Introduction

Over the past decade, Augmented Reality (AR) has gained huge momentum in both industry and academia. This concept became well-known to the general public thanks to the popular games on everyone's mobile phone, such as Pokémon Go (Niantic, San Francisco, CA), which was released in 2016 [197]. The leading Internet Technology companies, like Google (Mountain View, CA), Facebook (Menlo Park, CA), Apple (Cupertino, CA) and Microsoft (Redmond, WA), all started to invest in this technology and delivered software or hardware products to the market since a few years ago. Most smartphones nowadays are equipped with sensors and software frameworks to enable AR applications. In addition, Head-Mounted Displays (HMDs) started to appear in the consumer market, which were only affordable for military or research laboratories for decades. The availability of AR technologies have created many opportunities for fellow researchers, to allow for the implementation and

evaluation of innovative ideas about augmented reality.

Healthcare is one of the biggest industries in the United States. According to the U.S. Centers for Medicare & Medicaid Services, the national health spending is projected to grow in average 5.5% per year for 2018-2027, and reach nearly $6.0 trillion by 2027, which is 19.4% of the Gross Domestic Product (GDP) [175]. Augmented Reality (AR) has great potential in the healthcare industry, as it offers advanced visualization, that can be used for medical education [308], pathology demonstration, interventional assistance and etc.

This dissertation summarizes my works towards applying AR technologies for interventional assistance in image-guided surgery (IGS). In this chapter, introductory information about relevant technologies, technical and clinical challenges that we are facing, the thesis statement, outline and contributions are presented.

1.1 Augmented Reality (AR)

Broadly, AR is defined as “augmenting natural feedback to the operator with simulated cues" [163]. In order to clarify the connection and difference between various types of mixed reality displays, Milgram et al. proposed the “Reality-Virtuality Continuum", and defined AR and AV (Augmented Virtuality) based on whether the surrounding environment is principally real or virtual [163]. Later, Azuma et al. defined AR from a technical point of view; an AR system must have three charac-

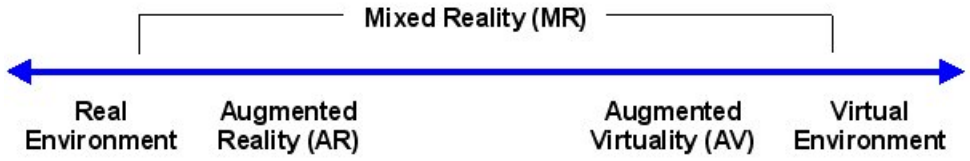


Figure 1.1: Reality-Virtuality continuum proposed by Milgram et al. [163]

teristics: 1) a combination of real and virtual views, 2) real-time interactions, and 3) views registered in 3D [16]. In this section, a brief history of AR is introduced, including the early developments, commercialization and recent advances.

Morton Heilig built the first immersive theater in the 1950s, named Sensorama [228]. At that time, Sensorama was viewed as the cinema of the future, which provides 3D stereoscopic color display, stereo sound, even wind and odor [102]. In 1968, Sutherland invented the world's first head-mounted display (HMD): The Sword of Damocles [248]. The HMD tracks the user's head via either an ultrasonic position sensor or mechanical linkage, and renders 3D lines that appear stationary in the room. This HMD was already a comprehensive system that continues to define the structure of today's devices. During the mid 1970s, Myron Krueger started to build Videospace, the first virtual environment comprised of video cameras and projectors [120]. Lintern et al. first applied and evaluated an AR system in the context of training landing skills of pilots [145]. Virtual landing cues are augmented on the aircraft simulator. In the early 1980s, with the increased popularity of television, AR started to appear on TV. For example, Dan Reitan overlaid weather radar images on images of the earth in weather broadcasts. In 1984, *The Terminator* movie demonstrated AR, in the form

of a heads-up display, for the general public. In 1989, George et al. developed an AR telescope, superimposing a star field to the view [78]. The term "Augmented Reality" was coined in 1990 by Boeing researchers, Thomas Caudell and David Mizell. During the 1990s, Feiner et al. proposed Knowledge-based Augmented Reality for Maintenance Assistance (KARMA) [65]. Researchers also started to use AR in the medical domain; Bajura et al. presented an AR system to visualize 3D ultrasound inside the body using an HMD [17]. Fuchs et al. incorporated an HMD for laparoscopic surgery [70]. Navab et al. presented the first deployment of AR in the operating room [176]. New concepts and tools are being created as well. Raskar et al. introduced Spatial Augmented Reality [214] in 1998. Hirokazu developed ARToolKit and used it in an augmented reality conferencing system [121] in 1999.

In the commercialization space, Sportsvision, acquired by SMT (Durham, NC) in 2016, first used AR to draw the 1st & Ten Yard line in an NFL game between the Ravens and the Bengals in 1998, which remains essential in football broadcasting to this day. In the 21st century, AR is becoming increasingly available to the general public, owing to significant improvements in hardware, sensing technologies (i.e., Microsoft Kinect [306]), software algorithms (i.e., SLAM [51]), iOS ARKit and Android ARCore, and reduction in cost. The mobile game, Pokémon Go, downloaded more than 500 million times in 2016, also popularized AR technology. Various powerful and affordable HMDs have been launched, including Google Glass and Microsoft HoloLens. In the 21st century, AR is becoming more and more readily available to

the general public. Researchers have been working on challenging topics to push the frontier of AR, e.g. SLAM [51], RGBD sensing and fusion [179]. Software platforms, such as Unity (San Francisco, CA) and Unreal Engine (Epic Games, Cary, NC), let developers create AR applications without much effort.

1.1.1 Augmented Reality Medium

As introduced in Sect. 1.1, AR requires a specialized hardware platform to blend the virtuality and the reality for the user, which is referred as the medium of AR. There are a few common types of AR mediums: monitor-based AR, projector-based AR, and HMD-based AR.

The monitor is a popular medium for AR, as it is an indispensable part of a computer. Monitors generally do not offer direct see-through capability. Therefore, on the monitors, the virtual content can be overlaid on the real information that is captured by a sensor, e.g. camera. Smartphone (or tablet) AR is a special kind of monitor-based AR, which is more portable than a monitor on the desktop. The mobile game, Pokémon Go, overlays virtual Pokémon characters on the phone-captured street view. An advantage of monitor-based AR is that the reality and the augmentation do not have relative latency because the computer has full control of the overall display. The reality can be digitally delayed to “wait for" the augmentation.

Projector-based AR is also known as spatial AR (SAR), where virtual objects are rendered directly within or on the user’s physical space [214]. Well-known projector-

based AR systems include the Sandbox developed at UC Davis [215] and the CAVE [47]. Using projectors, the augmentation is directly projected on the surface of objects. The users do not need to wear additional hardware for the AR experience.

In this dissertation, I present the works that are mainly using HMDs as the AR medium, which will be introduced in more detail in Sect. 1.1.2.

1.1.2 Head-Mounted Display (HMD)

As the name indicates, users wear the HMD on the head, and the virtual content is displayed on the HMD in front of their eyes. Based on the way that the reality is presented, HMDs can be further categorized as video see-through head-mounted display (VST-HMD) and optical see-through head-mounted display (OST-HMD).

With OST-HMD, computer generated graphics can be presented to the users while they are still able to see the real world through a semi-transparent display, while the VST-HMD blocks the user's direct view of the real world, but passes-through the reality that is captured by its sensors. OST-HMDs have various advantages over VST-HMDs, for instance, they are fail-safe for critical medical procedures [218]. Even if the virtual graphics fail to deliver, the user is still able to perform the procedure normally. Since the invention of HMD by Sutherland [248], engineers and scientists have been continuously pushing the frontier of this technology. After many years of being a high-end expensive piece of hardware, OST-HMDs have recently entered the consumer electronics market. Fig. 1.2 shows three recent OST-HMD products:

Figure 1.2: Example optical see-through head-mounted displays (OST-HMD)

Moverio BT200 by Epson (Suwa, Nagano, Japan), R-7 by Osterhout Design Group (San Francisco, CA) and HoloLens 1st gen by Microsoft (Redmond, WA).

In order to use HMDs for interventional assistance, it is critical to have fail-safe capability, therefore, we choose OST-HMDs over VST-HMDs for the research work presented in this dissertation. Specifically, I have used Epson Moverio BT200 and BT300, ODG R-7 and R-9, Microsoft HoloLens 1st and 2nd generation, and Magic Leap One (Plantation, FL). In Tab. 1.1 and Tab. 1.2, the hardware specifications of the above devices are listed.

Table 1.1: Hardware specifications of current OST-HMDs

Specs.	Moverio BT200	Moverio BT300	ODG R-7	ODG R-9
Optics	Projector LCD	Projector LCD	Projector	Projector
Binocular	✓	✓	✓	✓
Resolution	960×540	1280×720	1280×720	1920×1080
FOV	23° Diag.	23° Diag.	30° Diag.	50°+ Diag.
Computing	Pad	Pad	Onboard	Onboard
Processor	1.2GHz Dual	1.44GHz Quad	Snapdragon 805	Snapdragon 835
Memory	1GB RAM	2GB RAM	3GB RAM	6GB RAM
OS	Moverio OS	Moverio OS	Recticle OS	Recticle OS
SLAM	3-DOF	3-DOF	3-DOF	6-DOF
Eye Track.	✗	✗	✗	✗
Weight	88g	69g	182g	184g
Fixture	Glasses-like	Glasses-like	Glasses-like	Glasses-like
Interaction	Touchpad	Touchpad	Button, Touch	Button, Touch
Release	2014	2017	2017	2017
Price	$849	$700	$2750	$1800

Table 1.1 . . . continued

A	B			
Status	Discontinued	Unknown	Discontinued	Discontinued

Table 1.2: Hardware specifications of current OST-HMDs (continued)

Specs.	HoloLens 1	HoloLens 2	Magic Leap One
Optics	Waveguide	Waveguide	Waveguide
Binocular	✓	✓	✓
Resolution	1268×720	2048×1080	1280×960
FOV	$30° \times 17.5°$	$43° \times 29°$	$40° \times 30°$
Computing	Onboard	Onboard	Pad
Processor	1GHz CPU & HPU	Snapdragon 850 & HPU	NVIDIA Parker
Memory	2GB RAM	4GB RAM	8GB RAM
OS	Windows Holographic	Windows Holographic	Lumin OS
SLAM	6-DOF	6-DOF	6-DOF
Eye Track.	✗	✓	✓
Weight	579g	566g	345g
Fixture	Helmet-like	Helmet-like	Glasses-like
Interaction	Head, Hand (limited)	Hand, Eye	Controller
Release	2016	2019	2018
Price	$3000	$3500	$2295
Status	Discontinued	Shipping	Shipping

Six of the seven OST-HMDs listed in the above tables were released within the last 4 years. Four of them have discontinued the manufacture due to various reasons. ODG already went out of business. All of these indicate the fact that AR is a very rapid-moving industry, especially in the past few years.

1.2 Computer-Integrated Surgery (CIS)

Computer-Integrated Surgery (CIS) is a surgical concept, and is comprised of a set of methods. Computer technology has been widely used in various stages of

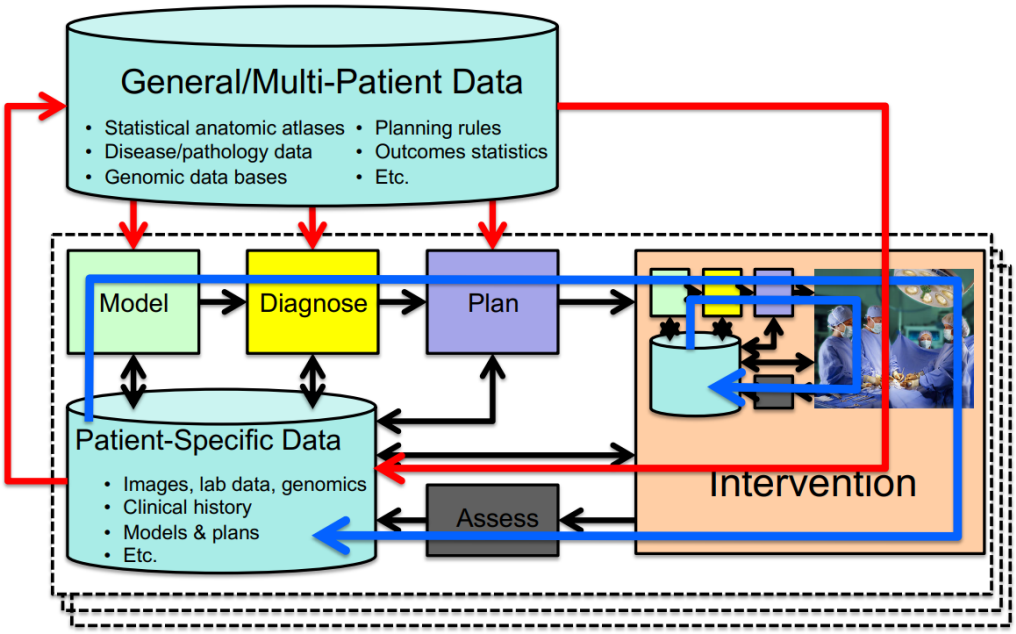


Figure 1.3: A diagram of computer-integrated interventional medicine (Figure courtesy Dr. Russell Taylor [253])

the surgical procedure, including patient modeling, diagnosis, preoperative planning, assessment, and navigation. Fig. 1.3 shows the overall workflow of CIS, which can also be called Computer-Integrated Interventional Medicine to reflect its use outside the operating room, such as in an interventional suite (credit to Prof. Russell Taylor, JHU course 601.455/655, Computer-Integrated Surgery) [255, 254].

CIS starts with the patient information, which includes medical images of the patient (e.g., computed tomography (CT), magnetic resonance imaging (MRI)), test results and other information. The **modeling** step within the CIS workflow combines the patient-specific information and general data, in order to better understand the patient information. After that, **diagnosis** of the patient and the interventional **plan** are made. A registration step is required to combine the patient model and surgical

plan with the actual patient. During the intervention, **assessment** of the operation is frequently made. Whenever additional imaging data and measurements are obtained, the information is used to update the patient model and the surgical plan. After the procedure is finished, the patient-specific information are gathered into the dataset, to aid further analysis and the overall understanding of the treatment. Therefore, a closed-loop surgical procedure is created.

1.3 Challenges

AR (especially HMD-based AR) is intrinsically an advanced human-computer interface and has many potential advantages when integrated with the CIS workflow to improve the perception of the surgical team.

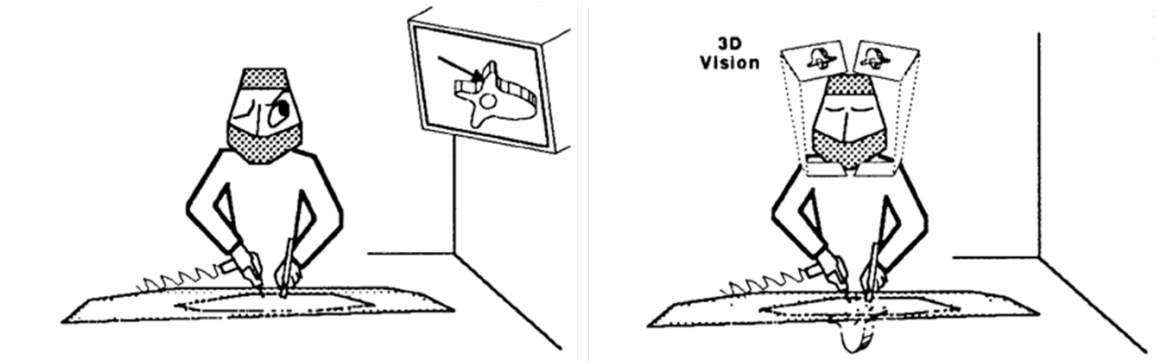


Figure 1.4: AR for scoliosis surgery proposed in 1995 (Peuchot et al. [195])

Fig. 1.4 illustrates the concept of using "Virtual Reality" (actually an AR approach) as an operative tool during scoliosis surgery [195]. Peuchot et al. proposed to superimpose a 3D transparent vision of the vertebra directly on the surgeon's op-

erative view. The expected advantages of such an approach include i) knowledge of the correction in the progress of a scoliotic curve, ii) safety during the placement of surgical implants, e.g. pedicular fixation, and iii) restriction in the extent of operative exposure [195].

However, nearly 25 years later, the “proposed" AR solution is still not a common practice in the OR because there are many challenges to realize the theoretical advantages of AR in the context of CIS. This section briefly discusses the challenges from both the technical and the clinical perspectives.

1.3.1 Technical Challenges

Despite the significant advancement of OST-HMD technology, there are still many limitations with the current generation of OST-HMDs. A perfect OST-HMD should be light, compact, offer wide field-of-view, high visual acuity of both virtual and real information, computationally powerful, and have a long battery-life. It is not possible to achieve all the above characteristics, therefore trade-offs broadly exist in the engineering of OST-HMDs. For example, an all-in-one helmet-like design is likely to be heavier and with less battery capacity. A tethered HMD benefits from the computational power of a workstation but is less portable.

In order to overlay virtual content with real-world objects, a display calibration procedure is required. The display calibration procedure is still time-consuming, inaccurate, and not user-friendly. It is a technical challenge to perform the display

calibration with minimum interaction and obtain high accuracy, which is especially critical for surgical applications. Chapter 2 describes our contributions to improve the display calibration procedure.

Another technical challenge for current OST-HMDs is the lack of optical magnification. Surgeons sometimes use microscopes or loupes to increase their visual acuity to observe fine details about the target anatomy. None of the existing OST-HMDs incorporates optical magnification capabilities and digital magnification currently provides lower image quality. Chapter 3 presents our solution to increase the visual acuity of the reality and virtuality based on an existing OST-HMD and loupes.

Further, the frame of OST-HMDs, which is not transparent, creates occlusion at the user's peripheral vision. Because the user's peripheral vision is critical to maintain safety and mobility, the occlusion caused by an OST-HMD will be a potential hazard for the OST-HMD user. In an operating room, the occlusion may reduce the surgeon's awareness about the surrounding environment, for example, the motion of the assistant. In more hazardous settings, such as those experienced by paramedics and combat medics, it may be life threatening. In Chapter 7, we present our methods to model the occlusion and the solutions to alleviate the issue.

There are many other technical challenges. For instance, the sensing technologies on the OST-HMD are still far from perfect, including eye tracking, self-localization and environment understanding.

1.3.2 Clinical Challenges

Before OST-HMDs and AR play an important role in surgeries, we (engineers, together with clinicians) need to prove that they provide actual clinical benefits. The very first step of which is to identify clinical use cases where OST-HMDs may provide benefits. After that, a specific application needs to be developed to validate the concept. The application needs to be integrated with the clinical workflow, having access to the medical data, relatively easy to set up, and compatible with clinical standards. The identification and design of such AR applications is the first clinical challenge. In this thesis, we develop a virtual monitor for image-guided surgery in Chapter 4, *ARssist* for robotic-assisted surgery in Chapter 5, and *ARAMIS* for minimally-invasive surgery in Chapter 6.

1.4 Thesis Statement

AR with OST-HMDs can be integrated with various clinical procedures to improve the surgical team's situation awareness, ergonomics and hand-eye coordination in some surgical tasks. Various prototypes based on current OST-HMDs and algorithms are developed to tackle the technical challenges of display calibration, occlusion at the periphery and lack of optical magnification.

1.5 Thesis Outline

This dissertation presents my work towards overcoming the aforementioned technical and clinical challenges to apply OST-HMD technologies for interventions. Chapter 2 introduces the importance of display calibration for accurate augmentation, and presents our innovative methods to make the display calibration more accurate, more robust and applicable. Chapter 3 presents our prototype to increase visual acuity of an OST-HMD, both for the reality and for the virtuality. We achieved this by integrating the OST-HMD with optical loupes, and a dedicated system calibration procedure to align the virtuality with the reality across the user's field of vision. Chapter 4 depicts an OST-HMD-based AR application for image-guided surgery, the "Virtual Monitor". We implemented a few visualization methods and applied it to percutaneous spine procedures. Appendix A presents a comprehensive literature review of AR application in robotic-assisted surgery (RAS), as background for the subsequent chapters. Chapter 5 introduces *ARssist*, which is an OST-HMD-based AR application for the patient-side assistant in RAS. *ARssist* shows significant improvement in hand-eye coordination and task safety, especially for inexperienced users and under difficult setups. Chapter 6 introduces *ARAMIS*, demonstrating the concept of "see-through surgery", where the laparoscopic surgeons can see the virtual augmentation of the patient internal anatomy. Chapter 7 illustrates the occlusion problem of current generation of OST-HMDs, and describes our method to alert OST-HMD users to potential danger in the environment. Chapter 8 concludes the thesis.

1.6 Thesis Contribution

The contributions of this dissertation are:

1. We develop a display calibration method for OST-HMD, fixed-head 2-DOF single point active alignment method (fh-SPAAM), to reduce the user related error. The method is presented and evaluated in Sect. 2.4. Alexander Winkler assisted me in the development and evaluation of the method.
2. We develop a display calibration method for stereoscopic OST-HMD, to model the physical property of the binocular system as additional constraints for the optimization, detailed in Sect. 2.5.
3. We develop a display calibration method for stereoscopic OST-HMD, to model the internal optical parameters, display parameters and virtual scene settings of OST-HMD as a black box, and then model the calibration as an end-to-end 3D-3D registration problem. The method integrates well with current development platforms of OST-HMD. Details are in Sect. 2.6. Ehsan Azimi contributed to the integration of the world-anchored tracking system, and assisted me in the evaluation of the method and paper writing.
4. We develop a prototype, AR-Loupe, integrating an OST-HMD (Magic Leap One) with an optical loupe, so that the user is able to have increased visual acuity of both the reality and the virtuality, with details in Sect. 3.4. We develop a system calibration algorithm for AR-Loupe, including interactive field-

of-vision segmentation and modified stereo-SPAAM to correctly provide overlay in the magnified and non-magnified field-of-vision, with details in Sect. 3.5. The occluded field-of-vision employs a novel method to ensure smooth transition between the magnified and non-magnified field-of-vision, via image warping on the display space. Tianyu Song developed the first version of the prototype as a course project for CIS II, under the supervision of me.

5. We develop a surgical AR application, "virtual monitor" in image-guided surgery, using OST-HMD and real-time medical image streaming. The "virtual monitor" supports various modes of visualization in the space of the operating room, catering to different clinical needs. The details are in Sect. 4.3. Dr. Bernhard Fuerst initiated the idea, and Dr. Mathias Unberath further refined the concept. Kevin Yu assisted in the development of the application.
6. We evaluate the application of the "virtual monitor" in percutaneous spine procedures with phantoms. The procedures include Vertebroplasty, Kyphoplasty, and Disc Decompression. The studies were mainly conducted by Dr. Mathias Unberath and Dr. Gerard Deib, detailed in Sect. 4.5.
7. We develop a set of criteria for evaluating OST-HMDs for the "virtual monitor" setup. The criteria include contrast perception, text readability, task load, frame rate and system lag. We use the criteria to compare three OST-HMDs, HoloLens 1st gen, Moverio BT-200 and ODG R-7, in Sect. 4.6. Dr. Bernhard Fuerst proposed the concept and contribution. Alexander Barthel contributed

in the evaluation, data analysis, and paper writing.

8. We develop *ARssist*, an OST-HMD based AR application for the bedside assistant in robotic-assisted surgery and evaluate the user performance during instrument insertion and tool manipulation for both experienced and inexperienced users. *ARssist* significantly improves the hand-eye coordination of the user, especially for less experienced users and in mis-orientation situations. Anton Deguet assisted me by developing software that provides low-latency UDP packet streaming from the da Vinci robot.
9. We develop *ARAMIS*, an OST-HMD based AR application for the laparoscopic surgeon, enabling "see-through surgery". The efficacy of *ARAMIS* is evaluated in a simulated peg transfer procedure. *ARAMIS* provides improved hand-eye coordination through the in-situ low-latency 3D visualization via point cloud. The bandwidth-efficient representation of the point cloud and utilizing GPU computing on the HoloLens to decode the point cloud are the keys to the low latency of *ARAMIS*. Xiran Zhang assisted me in developing the CUDA-accelerated disparity calculation from stereo endoscopic images.
10. We develop a method to restore the lost peripheral awareness caused by the occlusion of the hardware of OST-HMD. We model the occluded field-of-vision for a specific user and a specific OST-HMD, detailed in Sect. 7.4. We use LEDs or the screen edge of the OST-HMD as indicators of activity in the occluded field-of-vision. We calibrate the system so that specific indicators reflect the

change in the environment in specific directions. Dr. Alexander Plopski proposed to use LED indicators for this method, and contributed to the writing of the manuscript.

11. We conduct a systematic review of AR applications in robotic-assisted surgery and discuss the hardware components, application paradigm, clinical relevance and future perspectives in Appendix A. Jie Ying Wu assisted me in paper collection and manuscript writing.

1.7 Published Work

Materials from this dissertation appear in the following publications:

1. **Long Qian**, Jie Ying Wu, Simon DiMaio, Nassir Navab, Peter Kazanzides, "A Review of Augmented Reality in Robotic-Assisted Surgery," *IEEE Transactions on Medical Robotics and Bionics (TMRB), pp. 1-16. 2020.*

2. **Long Qian**, Xiran Zhang, Anton Deguet, Peter Kazanzides, "ARAMIS: Augmented Reality Assistance for Minimally Invasive Surgery Using a Head-Mounted Display," *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pp. 74-82. Springer. 2019

3. Rafa Rahman, Matthew Wood, **Long Qian**, Carrie Price, Alex Johnson, Greg Osgood, "Head-Mounted Display Use in Surgery: A Systematic Review," *Surgical Innovation (SRI)*. 2019

4. **Long Qian**, Anton Deguet, Peter Kazanzides, "dVRK-XR: Mixed Reality Extension for da Vinci Research Kit," *Hamlyn Symposium on Medical Robotics (HSMR)*, pp. 93-94. 2019.

5. **Long Qian**, Anton Deguet, Zerui Wang, Yun-hui Liu, Peter Kazanzides, "Augmented Reality Assisted Instrument Insertion and Tool Manipulation for the First Assistant in Robotic Surgery," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5173-5179. IEEE. 2019.

6. **Long Qian**, Alexander Plopski, Nassir Navab, Peter Kazanzides, "Restoring the Awareness in the Occluded Visual Field for Optical See-Through Head-Mounted Displays," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, Volume 24, Issue 11, pp. 2936-2946. IEEE. 2018.

7. **Long Qian**, Anton Deguet, Peter Kazanzides, "ARssist: Augmented Reality on a Head-Mounted Display for the First Assistant in Robotic Surgery," *Healthcare Technology Letters (HTL)*, Volume 5, Issue 5, pp. 194-200. IET. 2018.

8. Gerard Deib, Alex Johnson, Mathias Unberath, Kevin Yu, Sebastian Andress, **Long Qian**, Gregory Osgood, Nassir Navab, Ferdinand Hui, Philippe Gailloud, "Image Guided Percutaneous Spine Procedures using an Optical See-Through Head Mounted Display: Proof of Concept and Rationale," *Journal of Neurointerventional Surgery (JNIS)*, Volume 10, Issue 12, pp. 1187-1191. British Medical Journal Publishing Group. 2018.

9. **Long Qian**, Alexander Barthel, Alex Johnson, Greg Osgood, Peter Kazanzides, Nassir Navab, Bernhard Fuerst, "Comparison of Optical See-Through Head-Mounted Displays for Surgical Interventions with Object-Anchored 2D-Display," *International Journal of Computer Assisted Radiology and Surgery (IJCARS)*, Volume 12, Issue 6, pp. 901-910. Springer. 2017.

10. **Long Qian**, Ehsan Azimi, Nassir Navab, Peter Kazanzides, "Alignment of the Virtual Scene to the Tracking Space of a Mixed Reality Head-Mounted Display," *arXiv* 1703.05834. 2017.

11. **Long Qian**, Mathias Unberath, Kevin Yu, Bernhard Fuerst, Alex Johnson, Nassir Navab, Greg Osgood, "Towards Virtual Monitors for Image Guided Interventions Real-Time Streaming to Optical See-Through Head-Mounted Displays," *arXiv*, 1710.00808. 2017.

12. **Long Qian**, Alexander Winkler, Bernhard Fuerst, Peter Kazanzides, Nassir Navab, "Modeling Physical Structure as Additional Constraints for Stereoscopic Optical See-Through Head-Mounted Display Calibration," *IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 154-155. IEEE. 2016.

13. **Long Qian**, Alexander Winkler, Bernhard Fuerst, Peter Kazanzides, Nassir Navab, "Reduction of Interaction Space in Single Point Active Alignment Method for Optical See-Through Head-Mounted Display Calibration," *IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 156-157. IEEE. 2016.

Chapter 2

Display Calibration for OST-HMD

This chapter presents the research contributions related to the display calibration of OST-HMDs. We describe the background and challenges to create accurate AR visualization on OST-HMDs, and then present several innovative methods aiming to improve the ergonomics or accuracy of existing calibration methods, applied to the current generation of OST-HMDs.

2.1 Introduction

The immersion and efficacy of an AR application highly depends on how well virtual objects are superimposed into the real world. Fig. 2.1-left shows an example of poor alignment between the virtual content and real object that it should be registered with, and Fig. 2.1-right shows an example of good alignment. For a monitor-based AR

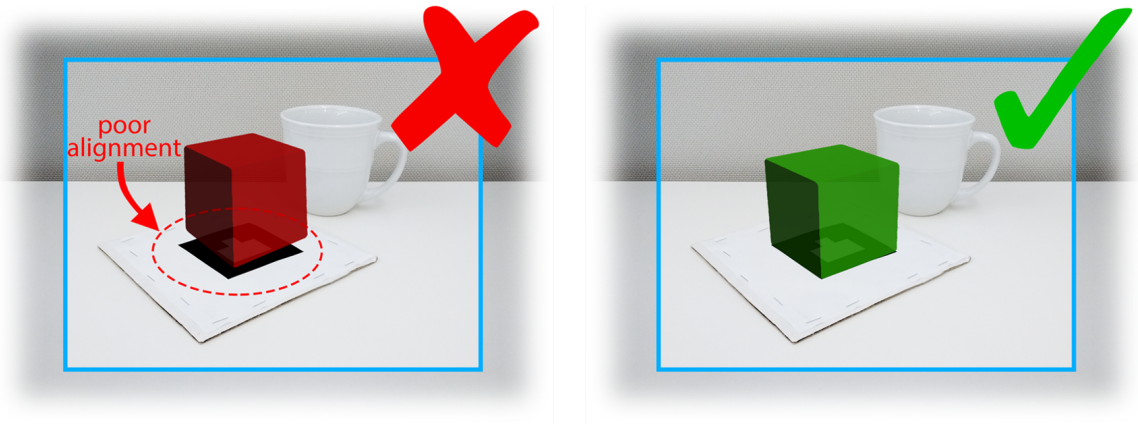


Figure 2.1: Example visualization on an OST-HMD with poor and good alignment

system, the alignment is performed via tracking and registration. Since the computer system has access to both the virtual content and image of the real environment, the alignment can be optimized and evaluated. However, with an OST-HMD, the user sees the reality with their own eyes; the "image" is projected on the user's retina. Hence, the computer system does not have access to what and where the user is exactly seeing. Therefore, it is a challenge for an OST-HMD to create precise alignment between the reality and the virtuality. A specific procedure, display calibration, is required to correctly offset the visualization [83].

Different OST-HMDs provide different levels of support to the display calibration, based on the target market and embedded sensing system. Epson Moverio BT-200 and BT-300, ODG R-7 and R-9 do not provide any display calibration support. In this case, the predominant use of them is limited to near-eye stereo displays. Microsoft HoloLens aims to create an in-room mixed reality experience. The indoor spatial mapping and self-localization on HoloLens is considerably stable and reliable. The

"Calibration" application shipped with the HoloLens, however, only calibrates the user's IPD. It is essentially part of the parameters for a stereoscopic OST-HMD, as presented in Sect. 2.5. For example, when the user with IPD $60\,mm$ is focused on an object at $0.5\,m$, an erroneous IPD value of $66mm$ (10% error) will cause the perceived depth of the object to offset by about $5\,mm$ (10% error). HoloLens does not offer accurate alignment after conducting the "Calibration". Magic Leap One not only supports in-room mixed reality application, but also offers an ImageTracking SDK which demonstrates the capability to superimpose virtual content on image markers [153]. The ImageTracking requires an eye tracking calibration (Magic Leap Visual Calibration) and the horizontal alignment of the stereoscopic screens. The accuracy of the alignment is $1.49\,mm$ (more details in Ch. 3). With HoloLens and Magic Leap One, it still requires additional efforts from the developers to accurately align virtual content with real objects.

A typical use case in surgical application that requires accurate alignment is the AR-based surgical guidance. Using OST-HMD, important guidance information, e.g. the location of the tumor, the planned trajectory of the instrument, can be overlaid registered with the patient anatomy. If the AR guidance showing the tumor is perceived with 5mm offset with respect to the real one, it is very likely to cause poor execution of the surgery plan or even lead to a medical error. The mis-alignment in a typical surgical guidance application with OST-HMDs can be decomposed into four components:

1. the tracking of target anatomy
2. the registration between the target anatomy and the virtual model space
3. the rendering of virtual content
4. the display calibration to correctly offset the visualization

The accuracy of tracking and registration is dependent on the embedded sensors, and the algorithms for the specific clinical use case. For example, rigid body 3D-3D registration is sufficient for orthopedic procedures, but anatomy deformation must be considered in the soft-tissue procedures [234]. The rendering of virtual content will introduce error in the alignment because the visualization is lagged compared to the reality, with additional movement of the human head. These are not in the scope of this thesis. In this chapter, we focus on reducing the amount of error caused by the display calibration.

2.2 Contributions

The contributions of this chapter are:

1. We develop a display calibration method for OST-HMD, fixed-head 2-DOF single point active alignment method (fh-SPAAM), to reduce the user related error. The method is presented and evaluated in Sect. 2.4. Alexander Winkler assisted me in the development and evaluation of the method.
2. We develop a display calibration method for stereoscopic OST-HMD, to model

the physical property of the binocular system as additional constraints for the optimization, detailed in Sect. 2.5.

3. We develop a display calibration method for stereoscopic OST-HMD, to model the internal optical parameters, display parameters and virtual scene settings of OST-HMD as a black box, and then model the calibration as an end-to-end 3D-3D registration problem. The method integrates well with current development platforms of OST-HMD. Details are in Sect. 2.6. Ehsan Azimi contributed to the integration of world-anchored tracking system, and assisted me in the evaluation of the method and paper writing.

2.3 Background and Literature Review

Researchers have developed various methods to perform the display calibration for OST-HMDs, addressing accuracy, robustness, and user-friendliness.

2.3.1 Single Point Active Alignment Method

The Single Point Active Alignment Method (SPAAM) is one of the widely applied display calibration methods due to its simplicity and accuracy [258]. In SPAAM, the human eye and the display screen is modeled as a virtual pinhole camera and the method solves the projection matrix of this virtual camera. A 3D point with known position (x, y, z) and a corresponding 2D point (u, v) on the screen are manually

collected. After a few 2D-3D point pairs are collected by the user, the 3×4 projection matrix G is calculated using Direct Linear Transform (DLT) [99].

More specifically, the 2D point set $\{(u^i, v^i),\ i \in [1, N]\}$ and 3D point set $\{(x^i, y^i,\ z^i),\ i \in [1, N]\}$ are both first normalized into $\{(\bar{u}^i, \bar{v}^i),\ i \in [1, N]\}$ and $\{(\bar{x}^i, \bar{y}^i, \bar{z}^i),\ i \in [1, N]\}$. Then we construct a $2N \times 12$ matrix B:

$$B = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \bar{x}^i & \bar{y}^i & \bar{z}^i & 1 & 0 & 0 & 0 & 0 & -\bar{u}^i\bar{x}^i & -\bar{u}^i\bar{y}^i & -\bar{u}^i\bar{z}^i & -\bar{u}^i \\ 0 & 0 & 0 & 0 & \bar{x}^i & \bar{y}^i & \bar{z}^i & 1 & -\bar{u}^i\bar{x}^i & -\bar{v}^i\bar{y}^i & -\bar{v}^i\bar{z}^i & -\bar{v}^i \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{2.1}$$

Essentially, DLT estimates a vector $\vec{p}$ that approximates $B\vec{p} = 0$, by calculating the eigenvector of B associated with the smallest eigenvalue. The projection matrix resulting from the DLT algorithm minimizes the total algebraic re-projection error.

Since the advent of SPAAM, many useful extensions based on it have been proposed. Genc et al. [76] introduced vision-based tracking of the 3D target for OST-HMD calibration to replace the magnetic-based tracking in the original work of SPAAM. Genc et al. also proposed *Easy-SPAAM* to take advantage of an existing calibration to reduce the required interactions [77]. It simplifies the interaction for users working frequently with the OST-HMDs. In [75], Genc et al. also proposed *Stereo-SPAAM* for efficient simultaneous calibration of two eyes. Grubert et al. attempted to collect multiple point correspondences from a single alignment [84]. The stylus-marker method [72, 73] was proposed independently of SPAAM, but it can still be thought of as a special case of SPAAM where 3D-2D point correspondences are collected and analyzed. In the stylus-marker method, the 3D point is the tracked

location of the user's finger.

2.3.2 Other Display Calibration Methods

Unlike modeling the mapping from 3D point sets to the 2D screen coordinates as a projection, the Display Relative Calibration [189] method takes the physical model of the optics into account, and separates the calibration into a user-independent offline calibration and a user-dependent calibration. Interaction-free calibration was introduced by Itoh et al., which utilizes eye-tracking technology to eliminate user active alignment [111]. Although the method frees the user from performing the tedious alignment task, the calibration accuracy at this time is limited [112].

Grubert et al. presented a complete survey of existing display calibration methods in 2017 [83]. More recently, data-driven non-parametric methods are proposed to tackle the display calibration problem, which no longer assume any model of the optical system [127, 235].

In this dissertation, SPAAM is chosen as the baseline calibration method because of its accuracy and wide application. Each of the improvements ([208, 209, 206]) is described in detail in the following sections.

2.4 Reduction of Interaction Space for Active Alignment

The accuracy of the display calibration of OST-HMD is subject to human-related errors, for example, postural sway [13], an unstable input medium [154], and fatigue. In this section, a new calibration approach is proposed: Fixed-head 2 degree-of-freedom (DOF) interaction for Single Point Active Alignment Method (fh-SPAAM) reduces the interaction space from a typical 6-DOF head motion to a 2-DOF cursor position on the semi-transparent screen. It uses a mouse as input medium, which is more intuitive and stable, and reduces user fatigue by simplifying and speeding up the calibration procedure.

2.4.1 Human-Related Error

Postural sway of the human deteriorates the alignment accuracy significantly [13]. It happens when the user is trying the stabilize the line of sight in order to make a precise alignment. However, the ability of human muscles to stabilize a static line of sight is limited [11, 12], and literature shows that this limitation affects the visual alignment precision under various task configurations while the user is acting as the operator. Researchers suggested that the compensation of body sway be enforced for better accuracy of the alignment task [12], but to the best of our knowledge this has not yet been presented or evaluated.

As a feature of the DLT method, the distribution of corresponding points, especially the distance from the user, has a large influence on the output [13]. Therefore, a good **coverage of depth** is critical to the quality of calibration of OST-HMDs. The 3D point of the corresponding point pair that the user collects should incorporate enough range of depth, instead of being concentrated at a few fixed depth locations. A specific depth pattern, the Magic Square sequence, is suggested in [14], which achieves a better result than the linearly sequential and static depth sampling strategy. When comparing the SPAAM calibration results for various target point distributions, the Magic Square sequence outperforms other sequences [169, 171]. If the user is allowed to perform alignment freely without following a good depth-coverage, he is not likely to follow the Magic Square depth sequence, thus the calibration result will not be satisfactory.

The activity of confirming an alignment usually requires a good eye-hand coordination, and involves a sudden movement of muscles, which can cause the alignment to shift. The proper choice of the **confirmation method** affects the amount of human-related error of the confirmation activity. Maier et al. [154] compare four different confirmation methods applicable for SPAAM: keyboard, hand-held, voice, and waiting. The comparison suggested a waiting period of 0.6 seconds provides the most accurate alignment, which also averages the user input and therefore effectively reduces the error caused by the limitations of the operator.

Fatigue of the user acts as another source of error during SPAAM calibration.

Although 6 alignments are sufficient for the DLT algorithm, usually 12 to 20 correspondences are collected in order to calibrate the projection matrix precisely [77]. Despite some efforts of wrapping up the calibration process into a user-friendly process, for instance in games [177], the calibration process remains dull and tedious [252]. As a result, it is important for the calibration system designers to reduce the time for completion, so that the users' focus and attention remains high throughout the calibration process.

2.4.2 Actuating Factor and Interaction Space

In order to facilitate the understanding of the problem, we define two concepts: the **actuating factor** and the **interaction space**.

The **actuating factor** of the display calibration process is defined as the mandatory difference between sequential alignments, which is usually predefined in the system. For SPAAM and the stylus-marker method [72], the actuating factor is defined to be the displacement of the crosshair on the foreground. The position of the crosshair is fixed, and it drives the user to react.

The **interaction space** is defined as the set of possible conditions of the input that the user can actively manipulate with respect to the actuating factor in order to collect 3D-2D correspondences. For each alignment, the input of the user is a configuration in interaction space. The interaction space for SPAAM is the 6-DOF head position and orientation, and for the stylus-marker method it is the position of

the user's finger (3-DOF) in tracking coordinates [73]. Each 3D-2D alignment in the process of OST-HMD calibration is the combination of one condition of the actuating factor and one sample of the interaction space.

2.4.3 Fixed-Head 2-DOF Interaction for SPAAM

Following the concept of **interaction space** and **actuating factor**, a new approach for performing SPAAM calibration is proposed: fixed-head 2 DOF interaction for SPAAM (fh-SPAAM), in which the user's head is fixed on a chin rest, 3D targets at different 3D poses are simulated and visualized on a screen, and the alignment of virtual and simulated real world targets is performed by manipulating a cursor instead of controlling the head pose (shown in Fig. 2.2). The interaction space for fh-SPAAM is the cursor position on the foreground, which is a two dimensional space.

2.4.4 Experiment

A comparative study with 16 participants was conducted to evaluate the users' alignment behavior for SPAAM and fh-SPAAM. The subjects were required to make 20 alignments in both methods under video see-through configuration, where the video captured by the HMD camera is displayed to the user to simulate OST-HMD calibration. For fh-SPAAM, the user controls the mouse and thus maintains a fixed head alignment. The user's clicked point is compared to the ground truth (the location

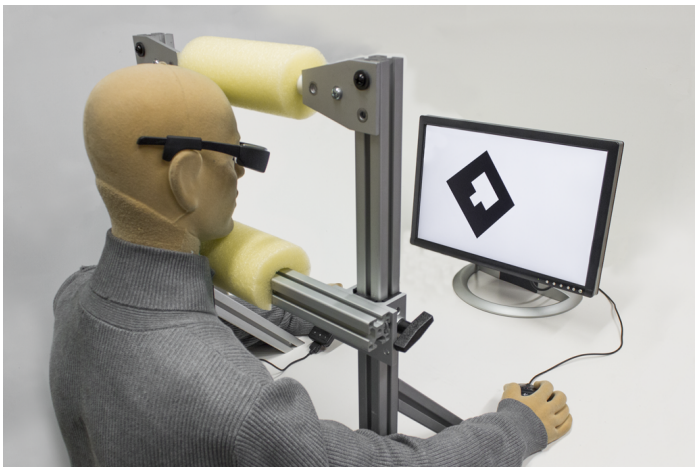


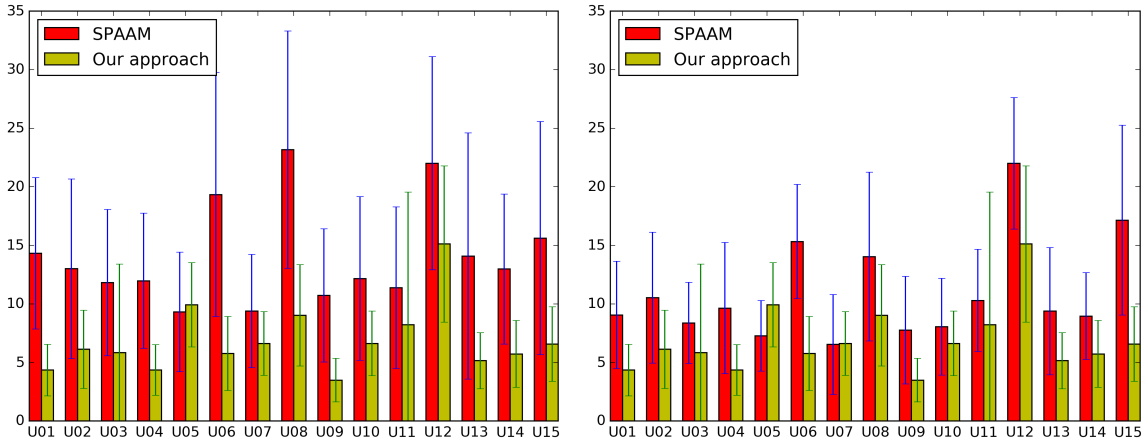
Figure 2.2: The setup of the fh-SPAAM. The user can position the head on a chin rest for additional stability, a screen is used to display perspective views of the marker images, and alignments are acquired using a computer mouse.

of the marker in the video). Subjective feedback about the system usability [19] was acquired from the user after the experiment.

One participant reported that the OST-HMD was not stable on top of prescription glasses, which led to inconsistent performance in the alignment exercise. Thus, the data of this participant was excluded from the study. An Epson Moverio BT-200 is used for the experiment. The vision-based tracking functionality is provided by ARToolKit [121].

2.4.5 Results and Discussion

Alignment error can be described as the Euclidean distance between the confirmed location and ground truth. To statistically compare the two experimental conditions, the users' mean value and standard deviation of the alignment errors are calculated



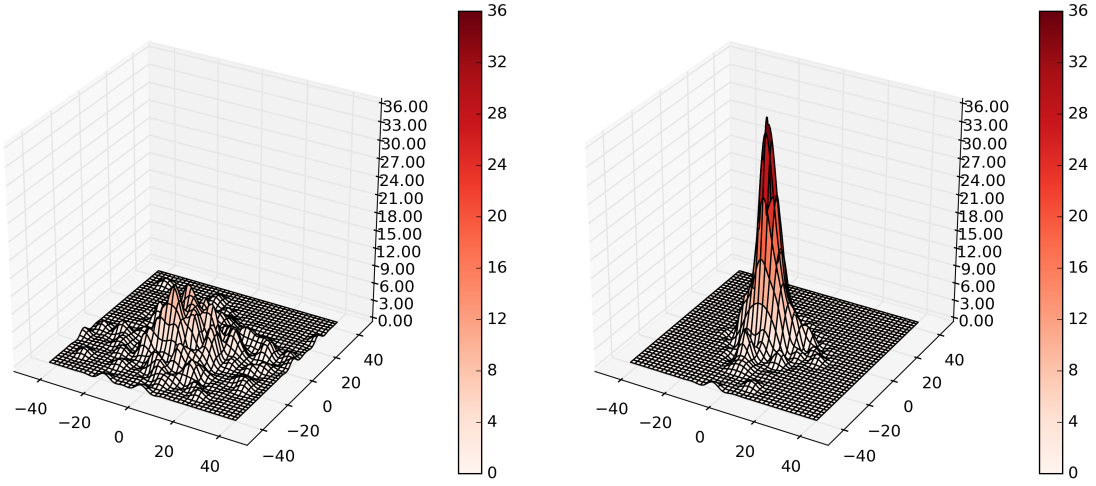
(a) SPAAM vs. fh-SPAAM **(b)** SPAAM (averaging) vs. fh-SPAAM

Figure 2.3: Mean alignment error in screen pixels for SPAAM and our approach for each participant. The whiskers show standard deviation. Our approach outperforms SPAAM as well as SPAAM with software based error reduction (averaging) in terms of accuracy and precision.

and analyzed. The data is illustrated in Fig. 2.3a, in which the means and standard deviations are plotted as bars and whiskers, respectively.

To test whether there is a statistically significant difference, we deploy the Kolmogorov-Smirnov (KS) test for the set of means to test normal distribution, which is a pre-condition to perform a t-test. The KS test results in a p-value greater than 0.05, indicating that the data is of normal distribution. A paired t-test comparing the mean alignment error of SPAAM and fh-SPAAM shows a p-value of 8.1×10^{-6}, revealing that the alignment error of our approach is significantly smaller compared to the alignments obtained with traditional SPAAM.

The comparison between fh-SPAAM and traditional SPAAM with averaging technique [258] is shown in Fig. 2.3b. Due to the constrained interaction space and less



(a) SPAAM **(b)** fh-SPAAM

Figure 2.4: Confirmation displacement in screen pixels of all alignments from the user study. The target is at (0,0) for both setups. Both accuracy and precision are improved by our approach.

error-prone input medium, user alignment error with fh-SPAAM is reduced significantly and consequently the OST-HMD calibration achieves higher accuracy. The fh-SPAAM method also reduced the time required for calibration by 40% when compared to the traditional SPAAM calibration (67.3s compared to 112.4s). With a mean System Usability Score of 72.3, the users assign our approach a "good" usability [19].

2.4.6 Summary

In this section, we described fh-SPAAM, which yields better calibration results than traditional SPAAM by limiting the user's interaction space and, therefore, reducing the user-related error during the active alignment procedure.

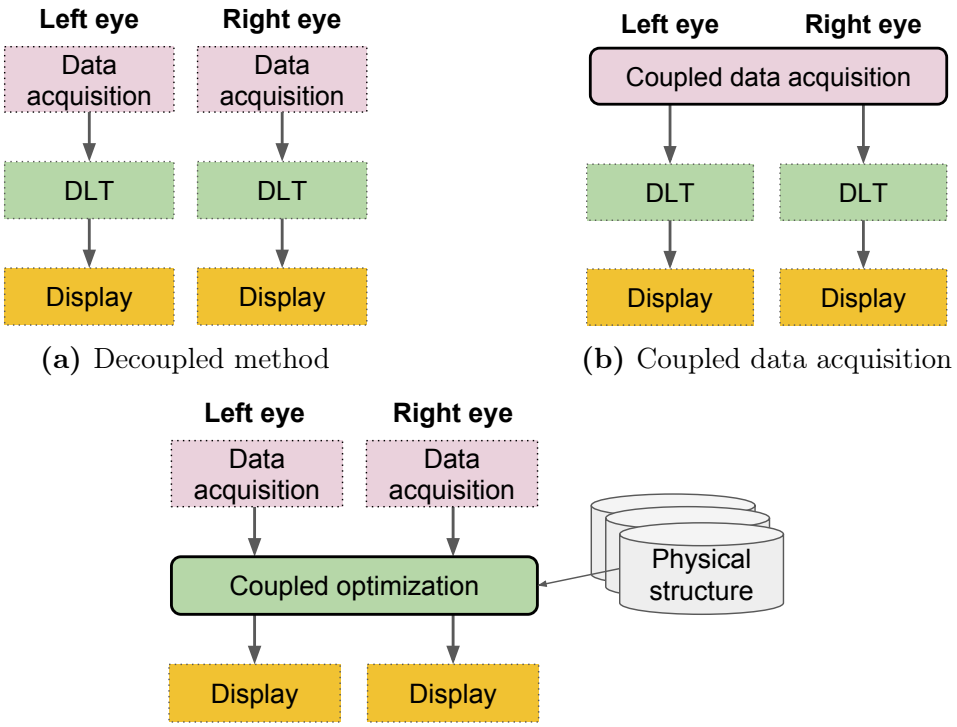
2.5 Additional Physical Constraints for Stereoscopic OST-HMD Calibration

For stereoscopic OST-HMD calibration, existing methods that calibrate both eyes at the same time highly depend on the user's unreliable depth perception. In addition, treating both eyes separately requires the user to perform twice the number of alignment tasks, and the calibration result does not necessarily satisfy the physical structure of the system. This section introduces a novel method that models physical structure as additional constraints and explicitly solves for the intrinsic and extrinsic parameters of the stereoscopic eye-camera system by optimizing a unified cost function. The calibration does not involve the unreliable depth alignment of the user, and lessens the burden for user interaction.

2.5.1 Motivation

For stereoscopic OST-HMD, there are two categories of calibration methods, illustrated in Fig. 2.5.

Decoupled methods treat both eyes individually: 3D-2D point pairs are collected separately for each eye. In this case, there is no coupling between the two eyes. Decoupled methods are based on the assumption that if the virtual object is aligned with the real-world target for both eyes, then humans can perceive the virtual display at the correct depth.



(a) Decoupled method

(b) Coupled data acquisition

(c) Coupled optimization

Figure 2.5: The workflow for different categories of calibration methods for stereoscopic OST-HMDs: (a) decoupled methods, (b) coupled data acquisition, and (c) coupled optimization (proposed).

Coupled data acquisition methods display the virtual object at a certain depth, and require the user to align it with its real counterpart at the correct depth. In each alignment task, the 3D-2D point pairs are collected for both left and right eyes. Stereo-SPAAM [75] falls into this category.

In this section, the **coupled optimization** method is proposed for calibrating stereoscopic OST-HMDs. The data acquisition stage of the coupled optimization method is the same as for the decoupled method. The mapping between the 3D-2D points are calculated by optimizing a cost function under various physical con-

straints. The introduction of these constraints in the optimization stage improves the consistency with the physical property of the stereoscopic system and, at the same time, reduces the parameter space, thus lessening the user interaction burden. Fig. 2.5 presents the workflow of the **decoupled method**, **coupled data acquisition method**, and **coupled optimization method**.

2.5.2 Physical Constraints for Stereoscopic OST-HMD

The **coupled optimization** method takes advantage of the effectiveness of SPAAM interaction [252], and overcomes the inaccuracy by adding physical constraints to the optimization problem.

By modeling the human eye and HMD screen as a virtual pinhole camera, 11 parameters [258] are required to describe the system. In decoupled methods, the degrees of freedom are doubled, however, by decomposition of the two projection matrices, the intrinsic and extrinsic parameters of the two eyes might not satisfy physical conditions, e.g. the interpupillary distance (IPD) calculated from the two extrinsic matrices may be different from the measured IPD.

In order to model physical properties explicitly in the state space, the parameters of the coupled optimization method are chosen as the intrinsic and extrinsic parameters of the projection matrix.

The relationship between the parameter space of the decoupled method and the

parameter space of the coupled optimization method is given by:

$$\begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & s & d_x \\ 0 & \alpha_y & d_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} R & T \end{bmatrix} \tag{2.2}$$

where $R = R(q_x, q_y, q_z, q_w)$ is the extrinsic rotation matrix represented by quaternions and $T = [t_x, t_y, t_z]'$ is the extrinsic translation vector. Therefore, the full parameter space of a stereoscopic OST-HMD is

$$\begin{aligned} \Psi = \{&\alpha_x^l, \alpha_y^l, s^l, d_x^l, d_y^l, q_x^l, q_y^l, q_z^l, q_w^l, t_x^l, t_y^l, t_z^l, \\ &\alpha_x^r, \alpha_y^r, s^r, d_x^r, d_y^r, q_x^r, q_y^r, q_z^r, q_w^r, t_x^r, t_y^r, t_z^r\} \end{aligned} \tag{2.3}$$

with additional constraints that the quaternions are unit quaternions.

Constraints on the parameter space are related to physical properties of the eyes and the OST-HMD screens:

- Pixel density of x and y axis on the screen is same, i.e. $\alpha_x^l = \alpha_y^l$
- Pixel density of the two virtual cameras is same, i.e. $\alpha_x^l = \alpha_x^r$
- There is no skew in user perceived image, i.e. $s^l = s^r = 0$
- Both eyes have the same viewing direction, i.e. $q^l = q^r$
- The line between both eyes is parallel to the horizontal image axis, i.e. $t_y^l = t_y^r$, $t_z^l = t_z^r$
- The interpupillary distance is given, i.e. $\left|t_x^l - t_x^r\right| = IPD$

These constraints can be used to reduce the state space directly, instead of handling constraints in the optimization stage via penalty functions.

The choice of constraints is dependent on the system and the application. For example, in order incorporate the constraint that both virtual cameras (eye-screen

virtual camera) are parallel, the developer needs to assume that the user is focused on a further distance. However, this might not be the case when the AR overlay appears at a close distance, e.g. within 20 cm. The method itself is flexible regarding to the choice of the set of additional constraints. It is left to the developer to choose the specific set of additional constraints. With more constraints applied to the calibration, fewer degrees of freedom remain in the system, and fewer numbers of alignment tasks are required.

2.5.3 Optimization

With intrinsic and extrinsic parameters explicitly expressed, the optimization problem is nonlinear, as shown in Eq. 2.2. Therefore, DLT is not applicable. We redefine the optimization problem here:

Problem Statement: Given 3D target positions in world coordinates $\{P_{Left}\}_i$ and $\{P_{Right}\}_j$, with corresponding 2D screen crosshair positions in pixel coordinates $\{I_{Left}\}_i$ and $\{I_{Right}\}_j$, $\Gamma(p, \theta)$ computes the projection of the 3D point on the screen with a projection matrix parameterized by $\theta \in \Psi$. The cost function $F(\theta)$ is the total reprojection error:

$$F(\theta) = \sum_i \left\| I^i_{Left} - \Gamma_L(P^i_{Left}, \theta) \right\| + \sum_j \left\| I^j_{Right} - \Gamma_R(P^j_{Right}, \theta) \right\| \tag{2.4}$$

The optimization problem is defined as: $\arg\min_\theta F(\theta), \theta \in \Psi$, which is a nonlinear problem where the physical structure is incorporated by reducing the dimension of the parameter space. Iterative methods, e.g. gradient descent, Newton's method,

Levenberg-Marquardt algorithm, can be used to calculate the parameters that result in the minimum reprojection error locally. Special attention should be paid to the quaternion parameters, which should be normalized after each iteration.

2.5.4 Experiment

A pilot study comparing the **decoupled method** and **coupled optimization method** is conducted. The user acquired 100 3D-2D point alignments for both eyes in the data acquisition stage, using Moverio BT-200. The 200 corresponding points are utilized in the separate DLT calculation and in the coupled optimization.

2.5.5 Results and Discussion

For the **decoupled method**, the decomposed intrinsic and extrinsic parameters are not consistent with the physical structure of the two eyes:

1. pixel density for both screens and both axes is different: $\alpha_x^l = 2637.88$, $\alpha_x^r = 2797.33$, $\alpha_y^l = 2506.21$, $\alpha_y^l = 2608.20$
2. skew factor is non-zero: $s^l = -95.69$, $s^r = 22.39$
3. rotation between the two virtual cameras is not identity: $q_x = -0.088$, $q_y = -0.066$, $q_z = -0.004$, $q_w = 0.994$
4. translation between the two eyes is not parallel to the horizontal image axis

The average reprojection error of the decoupled method is 6.211 pixels.

In the **coupled optimization method**, the physical constraints are strictly observed. The average reprojection error is 8.34 pixels. The reprojection error is larger than for the decoupled method because there are fewer number of free parameters, or more constraints in the coupled optimization method. As is shown in the above list, the calibration result of the decoupled method does not align with the physical reality.

An initial state vector $\theta_0 \in \Psi$ is required in coupled optimization. Due to the fact that the gradient descent method can find a local minimum, the initial value should be close enough to the actual value. The virtual camera formed by the user's eye and HMD screen is similar between different people, so it should be possible to use a nominal initial value for all calibrations based on coupled optimization.

Coupled optimization takes more time than DLT. However, since OST-HMD calibration is separate from the actual application, the time consumed by an iterative method (several seconds in the experiment) is not critical.

2.5.6 Summary

In this section, a new method is proposed that models physical structure as additional constraints for stereoscopic OST-HMD calibration. The **coupled optimization method** provides the advantage of the decoupled data acquisition and, at the same time, explicitly follows the physical requirement of the stereoscopic system.

2.6 Modeling Stereoscopic OST-HMD Calibration as 3D-3D Registration

In this section, we take one step further from Sect. 2.5, and consider the AR visualization pipeline as a "blackbox". We model the end-to-end calibration as a 3D-3D registration problem. The method integrates well with modern graphics engines and current generation of OST-HMDs.

2.6.1 Motivation: The "Blackbox"

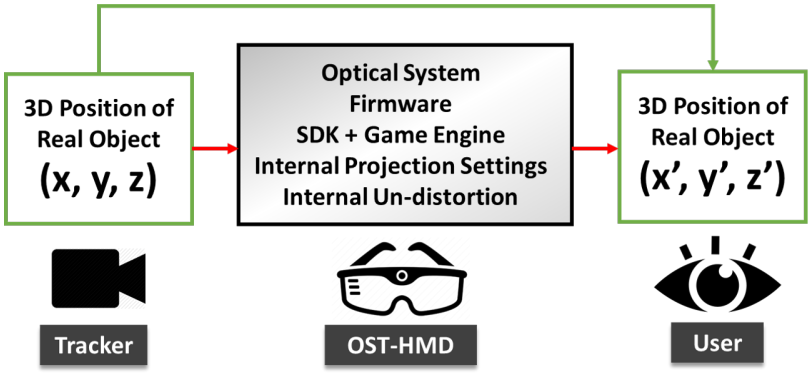


Figure 2.6: The concept of the "blackbox" for stereoscopic OST-HMD calibration

A typical OST-HMD-based AR application requires the display calibration to properly align the virtual content with the real counterpart, where the real object needs to be tracked by the OST-HMD. Therefore, the input information to the AR system is the 3D position of the object of interest, i.e. (x, y, z). On the other side, the display on the stereoscopic OST-HMD is generally rendered from a pair of virtual

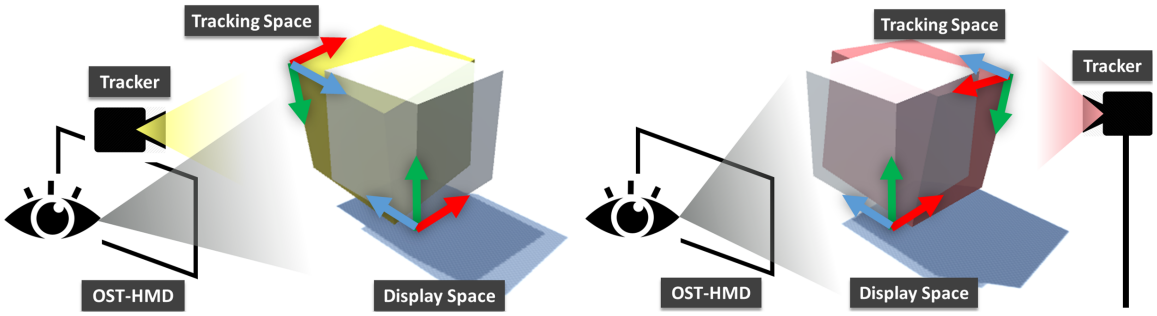
cameras in the 3D graphics scene. Essentially, the 3D position of the virtual object in the graphics scene, i.e. (x', y', z'), can be altered to adjust the rendering. The purpose of a display calibration method is to adjust the input 3D position of the real object (x, y, z), so that the output position that will be rendered by the stereo virtual cameras (x', y', z') aligns accurately with the real object. Fig. 2.6 illustrates the concept of the "blackbox" approach for stereoscopic OST-HMD calibration.

If we look into the "blackbox" between the input from tracking system, to the output in a virtual scene, many internal parameters and complicated processing are embedded in the stereoscopic OST-HMD. For rendering a virtual scene, the game engine usually uses internal projection parameters that are obtained from the manufacturer. The virtual cameras are separated with a proper distance that should match the user's interpupillary distance (IPD). The optical axis of the virtual cameras should be parallel. These internal settings actually guarantee the physical constraints proposed in Sect. 2.5. Once the stereo images are rendered, they are transmitted to be presented in front of the user via an optical system. The small optical components on the OST-HMD may create distortion or color aberration on the final image. Therefore, the 2D images are usually first artificially distorted or warped to offset the artifacts created by the optical system.

It is complicated to model and parameterize the visualization of an OST-HMD. However, since we do know the input and output for an AR application, we can model the internal process as a "blackbox" and estimate the behavior of this blackbox by

sampling some input and output data.

2.6.2 Tracking System



(a) Head-anchored tracking system **(b)** World-anchored tracking system

Figure 2.7: Two types of common tracking systems with HMDs

We first look into the input side of the "blackbox", the tracking system. AR based on OST-HMD usually incorporates two categories of tracking system: **head-anchored tracking system** (also called "inside-out") and **world-anchored tracking system** (also called "outside-in") [274].

All of the currently available OST-HMDs, including the devices listed in Tab. 1.1 and Tab. 1.2, have an embedded front-facing camera, which can serve as an optical tracker. A **head-anchored tracking system** has the advantage of providing similar line of sight to that of the user, but its performance is limited by the size, power consumption, and computational cost. Calibration of an OST-HMD using a head-mounted camera has been proposed [73, 76, 121]. Marker-based tracking algorithms have the advantage in simplicity and robustness [74, 121], and marker-free tracking algorithms offer better user experience [119, 179].

In a **world-anchored tracking system**, the pose of the tracker coordinate system remains unchanged with respect to the world coordinate system. Without the constraints imposed by power, computational resources and the type of technology used, a world-anchored tracking system can potentially be more accurate. Examples of world-anchored tracking systems include reflective markers tracker, electromagnetic sensing, and projective light-based tracker. World-anchored tracking systems are commonly used for VR headsets, e.g. Oculus

In this section, calibration of an OST-HMD based on a head-anchored tracking system and a world-anchored tracking system are both studied and presented.

2.6.3 End-to-End 3D-3D Registration

As shown in Fig. 2.6, if we model the complex optical system and internal parameters as a blackbox, we have both input and output of the blackbox to be 3D points in Euclidean space. We need to determine a transformation $T(\cdot)$ which maps 3D points from the world coordinates to a 3D virtual environment. Basically, if we are given the points $q_1, \cdots, q_n$, through the transform we observe $p_1, \cdots, p_n$ such that

$$p_i = T(q_i) \quad i = 1, \cdots, n. \tag{2.5}$$

We assume that both p_i and $q_i \in \mathbb{R}^3$. The goal is to estimate T based on a set of observations in the form of (q_i, p_i) for $i = 1, \cdots, n$. More specifically, the measurement of q_i is obtained from the tracking system, while the information of p_i is pre-defined and visualized on the OST-HMD. With the calculated transform $T(\cdot)$, a point from

the tracker coordinate system is mapped to that of the display coordinate system.

We further assume that the transformation $T(\cdot)$ is **linear**, and since our aim is to find the transformation between the 3D sensor tracking coordinate system and the 3D scene camera (visualization) coordinate system we assume that it is an **affine** transformation (12 unknown parameters), as the transformation between coordinate systems is affine. To verify this assumption, we also solve for the general case where the transformation T is a **perspective** transformation, with 15 unknown parameters (excluding an arbitrary scale parameter). In addition, because fewer unknown parameters require fewer calibration alignments and thus can considerably reduce the burden on the user, we also consider an **isometric** transformation that has 6 unknown parameters.

Different methods for solving these transformations have been studied [180, 105, 260, 170, 172]. Both **perspective** and **affine** transformation can be calculated with the Direct Linear Transformation (DLT) algorithm [99]. For an **isometric** transformation, the problem is equal to registration of two rigid 3D point sets; therefore, the absolute orientation method of Arun is used [10].

2.6.4 Implementation on Modern OST-HMDs

We used Microsoft HoloLens 1st gen and Epson Moverio BT-300 to implement our calibration method. For the HoloLens, both head-anchored and world-anchored tracking systems are studied for the calibration. For Moverio BT-300, the head-

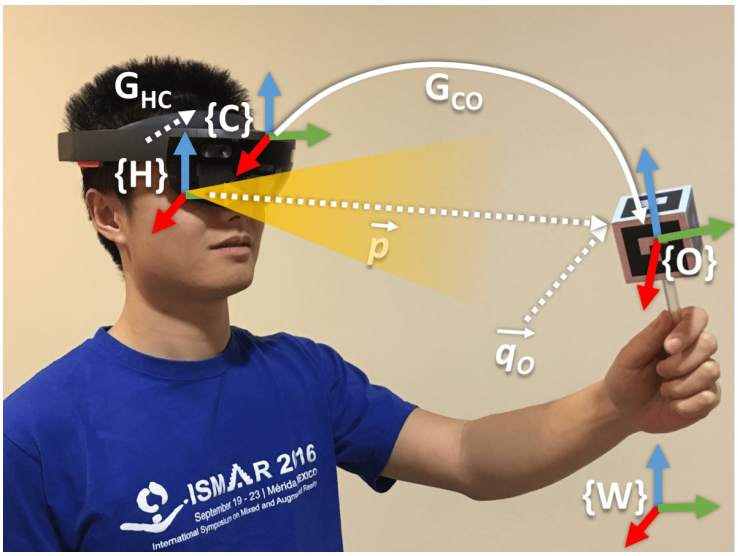


Figure 2.8: Implementation of the calibration with Microsoft HoloLens and head-anchored tracking system. The HoloLens embedded front-facing RGB camera is used as the head-anchored tracker.

anchored tracking system is studied.

2.6.4.1 HoloLens with Head-Anchored Tracker

The embedded front-facing RGB camera of HoloLens is used as the head-anchored tracker. Fiducial markers are attached to a real object that is held by the user. The coordinate systems of the tracker, object and OST-HMD are represented as $\{C\}$, $\{O\}$ and $\{H\}$, respectively, as shown in Fig. 2.8. Since the camera is rigidly mounted on the OST-HMD, the extrinsic geometric transformation between the camera and the HoloLens G_{HC} is fixed. The point for alignment is fixed at $\vec{q_O}$ with respect to the coordinate system of $\{O\}$. Its corresponding virtual point is at $\vec{p}$ in the HMD display coordinate system $\{H\}$. The pose of the tracked object G_{CO} is determined

with a marker-tracking package HoloLensARToolKit at runtime. The details of the HoloLensARToolKit project will be presented in Sect. 2.6.8. Eventually, the point sets $\{q \,|\, q_i = G_{CO,i} \cdot \vec{q_O},\, i = 1, \cdots, n\}$ and $\{p_i \,|\, i = 1, \cdots, n\}$ are used for the OST-HMD calibration described in Section 2.6.3.

2.6.4.2 Moverio BT-300 with Head-Anchored Tracker

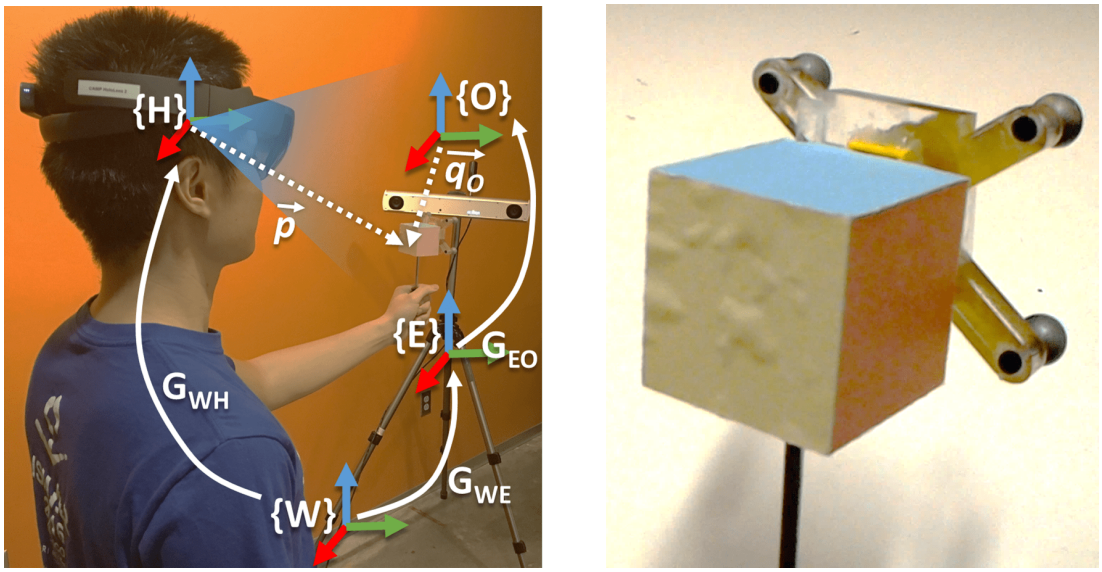
Similar to the HoloLens, we implement our calibration method on Moverio BT-300, using its embedded front-facing RGB camera for tracking. The user holds the same object with attached ARToolKit marker for tracking.

We treat the BT-300 as a "blackbox" and do not perform any display calibration prior to applying our method. The default projection matrix of BT-300 is used, which causes poor initial alignment of virtual objects. We intend to keep the default projection matrix in the "blackbox" and evaluate whether our calibration method can overcome the initial setting.

2.6.4.3 HoloLens with World-Anchored Tracker

We use Atracsys FusionTrack 500 (Puidoux, Switzerland) as the world-anchored tracking system. Passive spherical markers compose a frame which is attached to the cube that is used for alignment (Fig. 2.9b).

As shown in Fig. 2.9a, the coordinate systems of the world-anchored tracker, object, OST-HMD and world are represented as $\{E\}$, $\{O\}$, $\{H\}$ and $\{W\}$. It should



(a) Transformation graph **(b)** Tracked object

Figure 2.9: Implementation of the calibration with Microsoft HoloLens and world-anchored tracking system (Atracsys FusionTrack 500). Passive spherical markers form a frame that is attached to the colored cube for tracking.

be noted that both $\{C\}$ and $\{E\}$ represent the tracker coordinate system. However, since our general workflow is different for these two tracking systems, we refer to the HMD camera as $\{C\}$ and external tracker as $\{E\}$ to reduce ambiguity when explaining both methods. The main conceptual difference between head-anchored and world-anchored tracking systems for our calibration process is as follows.

In the head-mounted tracker case, the transformation G_{HC} between the tracker $\{C\}$ and the OST-HMD display $\{H\}$ is fixed, but this is not the case for the world-anchored tracker, where the transformation G_{HE} is expressed as $G_{HE} = {G_{WH}}^{-1} \cdot G_{WE}$. Since the world-anchored tracker does not change its pose in the room, G_{WE} is fixed. Therefore, an extra component is needed to maintain and update the transformation G_{WH} between the world and HMD display $\{H\}$, so that the transformation between

the tracker and the display G_{HE} can be determined. The SLAM-based spatial mapping capability of the HoloLens fills this gap and completes our transformation chain from the tracked object to the user's view.

2.6.5 Experiment

To analyze and evaluate our proposed calibration method, experiments were carried out for each implementation of Sect. 2.6.4. 20 trials of calibration and evaluation were conducted.

2.6.5.1 Experiment Procedure

The calibration workflow diagrams for all three implementations are depicted in Fig. 2.10.

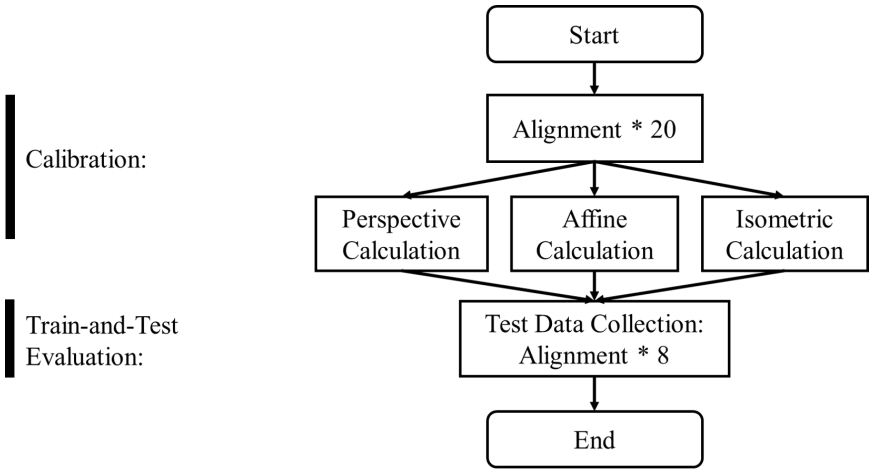
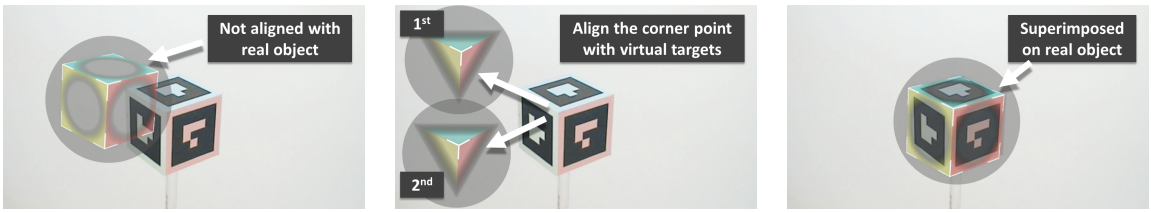


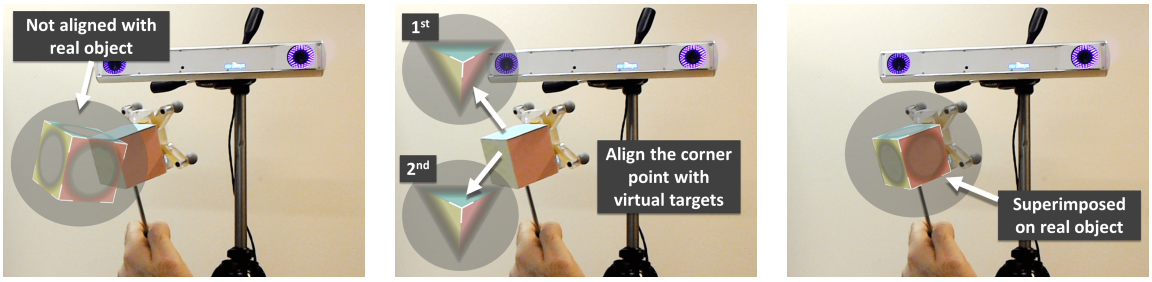
Figure 2.10: The overall experiment procedure for all three implementations

First, as represented in Figs. 2.8 and 2.9a, the user wears the OST-HMD (HoloLens



(a) Before calibration **(b)** Making alignments **(c)** After calibration

Figure 2.11: Graphical illustration of the experiment procedure with head-anchored tracking system



(a) Before calibration **(b)** Making alignments **(c)** After calibration

Figure 2.12: Graphical illustration of the experiment procedure with world-anchored tracking system

or Moverio BT-300) and is given a real object (cube) for alignment. Before calibration, the virtual overlay is not correctly aligned with the tracked cube, as shown in Figs. 2.11a and 2.12a. Using automated voice commands, the user is then instructed to perform the calibration step by step.

First, a virtual cube is displayed and the user should try to align only one corner of the cube with its real counterpart in her/his hand (Figs. 2.11b and 2.12b). Once the user is satisfied with the alignment, a button is clicked for the confirmation. Only the corner position is measured for the alignment. The colored faces of the virtual and real cubes make the alignment task more intuitive with the additional depth cue and color similarity. Next, the virtual cube appears in another location in the

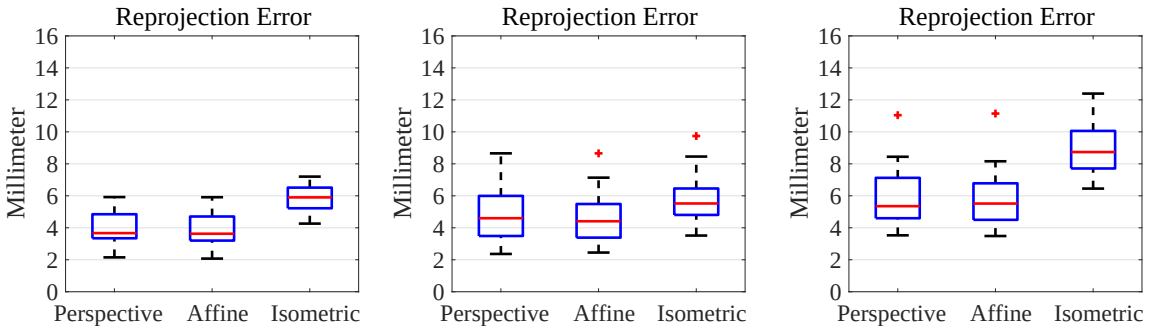
user's field of view. We try to cover the entire workspace within the reach of the user so that our calibration results are balanced and less biased towards a certain geometrical location. This process continues until 20 points are collected. At this point, the affine, perspective and isometric 3D projection matrices are calculated with their corresponding reprojection errors.

2.6.5.2 Experiment Evaluation

Evaluating an OST-HMD calibration has always been challenging because only the user wearing it can observe the superimposed objects that result from the calibration. **Train-and-test** is a standard approach where evaluation is performed with additional samples that were not used for the solving the calibration. Specifically, the user is asked to collect 8 additional samples, and these samples are tested against the calibration calculated with the data sets consisting of the 20 alignments. Reprojection error of the test data is computed based on each of the three transformation matrices (perspective, affine, isometric).

2.6.6 Results and Discussion

Two users familiar with various OST-HMD systems and calibration techniques performed the experiment with each implementation for 10 times. In total, 20 sample data are collected per method per setup. The results and error analysis are presented in this subsection.



(a) HoloLens w. Head **(b)** Moverio w. Head **(c)** HoloLens w. World

Figure 2.13: Evaluation results of train-and-test, for the three implementations, each with three different geometrical models ($N = 20$).

Table 2.1: Evaluation results of train-and-test ($N = 20$), corresponding to Fig. 2.13. The unit is millimeter, and data is presented as mean ± std. The smallest number for each experiment setup is highlighted using bold font.

Models	Perspective	Affine	Isometric
HoloLens w. Head	4.04 ± 1.04	$\mathbf{3.96} \pm 1.06$	$5.86 \pm \mathbf{0.80}$
Moverio w. Head	4.75 ± 1.63	$\mathbf{4.60} \pm \mathbf{1.55}$	5.76 ± 1.67
HoloLens w. World	5.88 ± 1.81	$\mathbf{5.83} \pm 1.78$	$8.92 \pm \mathbf{1.60}$

2.6.6.1 HoloLens with Head-Anchored Tracker

Fig. 2.13a depicts the reprojection error of the testing data for the HoloLens using the calibration results from perspective, affine and isometric transformation matrices. The mean and standard deviation of the reprojection error are: $4.04 \pm 1.04\,mm$ for perspective model, $3.96 \pm 1.06\,mm$ for affine model, and $5.86 \pm 0.80\,mm$ for isometric model, as shown in Tab. 2.1.

We use Mann–Whitney U test (Wilcoxon rank sum test) to compare different geometrical models in a pair-wise manner. Mann–Whitney U test is an unpaired

non-parametric test with null hypothesis that the distributions of two populations are equal [156]. In other words, a rejected null hypothesis means that the two distributions are significantly different. A p-value < 0.05 is considered statistically significant.

The p-values for each pair-wise comparison are: $p = 9.75 \times 10^{-6}$ comparing perspective and isometric models, $p = 7.58 \times 10^{-6}$ comparing affine and isometric models, $p = 0.69$ comparing affine and perspective models. Both perspective and affine models are significantly better than the isometric model in terms of the reprojection error, but the difference between the perspective and affine models is not significant.

2.6.6.2 Moverio BT-300 with Head-Anchored Tracker

The mean and standard deviation of the calibration error of each geometrical model are: $4.75 \pm 1.63\, mm$ for perspective transformation, $4.60 \pm 1.55\, mm$ for affine transformation, and $5.76 \pm 1.67\, mm$ for isometric transformation (as shown in Tab. 2.1). Affine transformation yields smaller alignment error, in the sense of both average value and standard deviation. Similar to the previous subsection, we use Mann–Whitney U test to study whether the error for different geometric models comes from different distributions without the normality assumption. The probability that the null hypothesis is rejected is determined to be $p = 1.93 \times 10^{-2}$ comparing the isometric and affine models, $p = 4.99 \times 10^{-2}$ comparing isometric and perspective models, and $p = 0.76$ comparing affine and perspective models. The statistical results are exactly the same as the previous subsection. Both perspective and affine models are signif-

icantly better than the isometric model in terms of the reprojection error, but the difference between the perspective and affine models is not significant.

2.6.6.3 HoloLens with World-Anchored Tracker

The reprojection error of the calibration results applied on the testing dataset is shown in Fig. 2.13c and Tab. 2.1. For perspective transformation, the mean and standard deviation of the reprojection error is $5.88 \pm 1.81\, mm$, while the affine transformation yields an error of $5.83 \pm 1.78\, mm$ and the isometric transformation yields an error of $8.92 \pm 1.60\, mm$. We use Mann–Whitney U test to study whether the error for different geometric models comes from different distributions without the normality assumption. The probability that the null hypothesis is rejected is determined to be $p = 8.60 \times 10^{-6}$ comparing the isometric and affine models, $p = 1.41 \times 10^{-5}$ comparing isometric and perspective models, and $p = 0.97$ comparing affine and perspective models. The statistical results are exactly the same as the previous two subsections. Both perspective and affine models are significantly better than the isometric model in terms of the reprojection error, but the difference between the perspective and affine models is not significant.

From the above experimental results, we can observe that both the affine model and the perspective model are able to capture the 3D-3D mapping, significantly better than the isometric model. The difference between the affine model and the perspective model is not significant in all three cases. Therefore, the affine model may be

more suitable to be used as the underlying model to solve for the 3D-3D registration problem, because the additional degrees-of-freedom of the perspective model do not reduce the calibration error. The perspective model may be over-fitting the display calibration.

2.6.7 Summary

In this section, we proposed a "blackbox" approach for solving the transformation between the tracking coordinate system and the virtual scene coordinate system. We applied our method for calibration of OST-HMDs, using both head-anchored and world-anchored tracking systems, and using affine, perspective and isometric transformation models. Experimental results indicated that the affine model better captures the underlying 3D-3D mapping. The results validated our hypothesis to model the OST-HMD complex visualization system as a "blackbox" and calibrate it end-to-end.

2.6.8 Open Source Contribution: HoloLensARToolKit

In order to use the HoloLens front-facing camera as an optical tracker, I developed HoloLensARToolKit and made it open source on GitHub under LGPL v3.0 license. HoloLensARToolKit is built on top of ARToolKit, which is a popular open source AR package developed in the 1990s [121]. ARToolKit is also using LGPL v3.0 license.

The main contributions of HoloLensARToolKit are:

- A C/C++ wrapper of ARToolKit that is compatible with Universal Windows Platform (UWP)
- Interface with HoloLens camera API to fetch locatable camera images
- Multithreading solution to off-load the heavy computation from the main rendering thread
- Provide Unity samples of tracking with barcode, pattern, multi-barcode markers, to allow developers to easily extend the implementation.

The current implementation widely uses the .NET Task-based Asynchronous Pattern to parallelize video capture, tracking, and Unity rendering. The dependency between each module is loosened. HoloLensARToolKit is able to achieve: rendering at 45-60 fps, video capture at 30 fps, and tracking at 25-30 fps performance. A new development branch *feature-grayscale* further improves the performance by only parsing the grayscale part of the NV12 image from the HoloLens camera. It saves the time of color conversion and copying of a large buffer. Currently, the open source package has received 190 stars and 57 forks on Github.

2.7 Closing Remarks

This chapter presents our effort towards improving the display calibration of OST-HMDs.

Firstly, we invented a method (fh-SPAAM) that focuses on improving the user ergonomics, which consequently improves the accuracy of calibration. The user's interaction space in fh-SPAAM is limited to 2-DOF, and eliminates a large extent of user alignment error.

Secondly, we considered to incorporate physical constraints in the calibration, especially for stereo OST-HMD. Many nominal constraints exist when calibrating stereo OST-HMD, for example, the extrinsic parameters of both pinhole cameras (eye-screen) should match the relative transformation between the user's actual eyes. These constraints are not taken into account for SPAAM-based methods, which directly estimate elements of the projection matrix. We created our calibration model and solution, which is validated with a pilot user study.

Lastly, we took one step further, and modeled the visualization system of stereo OST-HMD as a "blackbox", whose input is 3D position provided by a tracking system (either head-anchored or world-anchored) and the output is another 3D position of the augmentation in the virtual graphic scene. We assume this "blackbox" is linear since both input and output are 3D vectors in the Euclidean space. We propose to use affine, perspective and isometric transformations to model the linear system. The method was implemented with both Microsoft HoloLens and Moverio BT-300 and validated with experiments.

Each of the developed methods has its limitations. fh-SPAAM requires the user to sit down with a chin-rest, which is not suitable for daily applications or for wide usage.

The second method considering physical constraints involves an iterative optimization solution, which suffers from local minima and longer time of convergence. Practically speaking, the blackbox approach is easier to be integrated with existing AR/VR development platforms like Unity. The game engines create a 3D virtual environment to be visualized. The internal parameters of the "blackbox" is good enough to deliver 3D virtual content in the front of the user, with some spatial offset. The blackbox approach can minimize this offset. I choose to use the blackbox approach as the initial calibration for the majority of the clinical applications in this thesis, e.g., *ARssist* and *ARAMIS*, which will be introduced in Ch. 5 and Ch. 6. When the offset is noticeable, I manually offset the position to have better alignment, due to the time limitation during the user studies. Also, all the three methods introduced in this chapter are based on the assumption or model of the Single Point Active Alignment Method (SPAAM), e.g., pinhole camera model without distortion.

Although the methods have shown promising improvement, they are still far from being used for daily application because they still take quite a large amount of time and cannot be "recycled" for another person or another use. From the perspective of user-friendliness, an interaction-free method is definitely more interesting for AR users. OST-HMD manufacturers started to build extra sensors into the devices, with the intention to ease the calibration procedure. HoloLens second generation and Magic Leap One both contain embedded eye-trackers. The display calibration of OST-HMD is a fast-moving field. I believe the community is not far from completely

solving the display calibration problem.

2.8 Published Work

Materials from this chapter appear in the following publications:

1. **Long Qian**, Ehsan Azimi, Nassir Navab, Peter Kazanzides, "Alignment of the Virtual Scene to the Tracking Space of a Mixed Reality Head-Mounted Display," *arXiv* 1703.05834. 2017.

2. **Long Qian**, Alexander Winkler, Bernhard Fuerst, Peter Kazanzides, Nassir Navab, "Modeling Physical Structure as Additional Constraints for Stereoscopic Optical See-Through Head-Mounted Display Calibration," *IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 154-155. IEEE. 2016.

3. **Long Qian**, Alexander Winkler, Bernhard Fuerst, Peter Kazanzides, Nassir Navab, "Reduction of Interaction Space in Single Point Active Alignment Method for Optical See-Through Head-Mounted Display Calibration," *IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 156-157. IEEE. 2016.

Chapter 3

AR-Loupe: Zoomable AR with OST-HMD and Loupe

This chapter presents the research contributions related to improving the visual acuity of both real and virtual content with an OST-HMD. Optical zoom is provided via an attached binocular loupe, and the digital zoom of the virtual content is provided by a user-specific calibration procedure. The prototype is named "AR-Loupe" and is demonstrated in Fig. 3.1.

3.1 Introduction

In the previous chapter, we have discussed the requirement for display calibration of OST-HMDs. Another limitation of the current generation of OST-HMDs is that

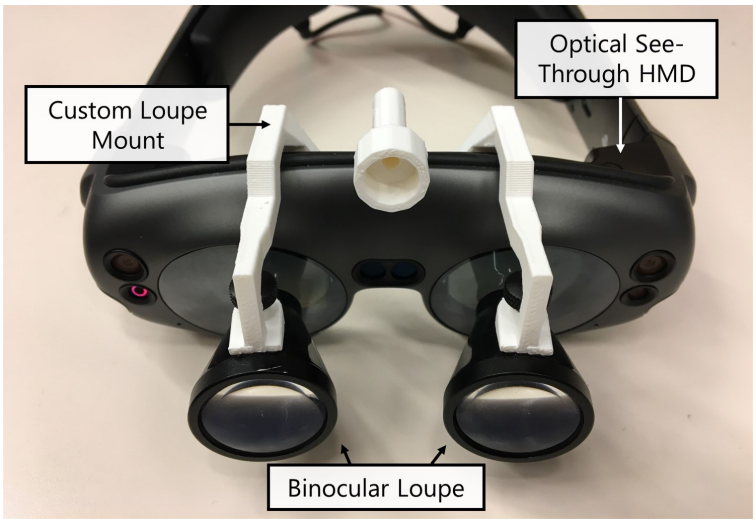


Figure 3.1: The hardware design of AR-Loupe

they are not able to enhance the user's natural vision. In other words, OST-HMDs cannot increase the user's visual acuity of the real world. For example, when a dentist is treating a patient with tooth decay, an OST-HMD is able to display other relative information to the dentist, e.g. a 3D model, but it cannot increase the dentist's capability to observe the cavities with better clarity.

On the other hand, people have been wearing eyeglasses as a vision aid for hundreds of years [219]. For certain critical tasks that require high visual acuity, people use loupes or microscopes to magnify the object so they can operate with better precision. In oculoplastics, a survey among oculoplastic surgeons in North America revealed that 95% of the survey respondents owned loupes and 78% regularly used them [279]. However, AR cannot be easily achieved with head-mounted loupes because i) the users are already wearing a visual aid, and ii) due to the difficulty in calibrating the virtual content with the magnified real-world object.

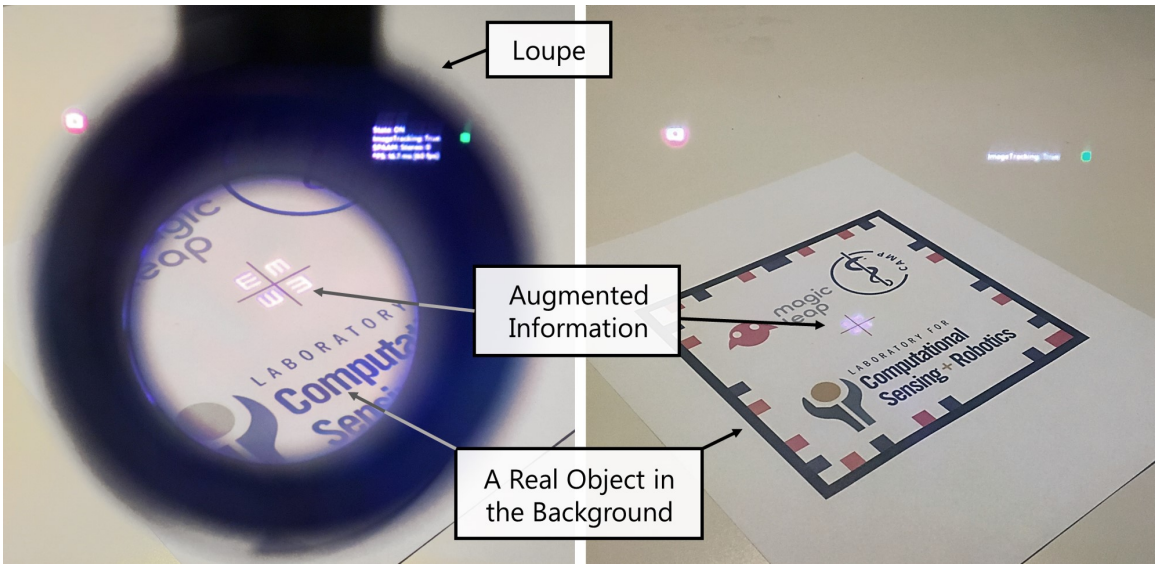


Figure 3.2: The see-through view with AR-Loupe, where both the reality and the virtuality are magnified.

In this chapter, we propose the concept of zoomable augmented reality, which has two main characteristics:

- the reality can be magnified or minified in, or part of, the user's field-of-vision.
- the augmentation appears registered with the reality across the user's field-of-vision, including the magnified (minified) portion and the normal portion.

We develop the hardware prototype, AR-Loupe, by integrating an OST-HMD (Magic Leap One) and a binocular loupe, with proper calibration and visualization methods, to achieve zoomable AR. Outside of the magnified area, normal AR is still available. We also propose methods to visualize occluded information due to the hardware occlusion and light refraction. Therefore, AR is available across the field-of-view. The transition between the magnified and the non-magnified field-of-vision is smooth.

3.2 Contributions

The contribution of this chapter is:

1. We develop a prototype, AR-Loupe, integrating an OST-HMD (Magic Leap One) with an optical loupe, so that the user is able to have increased visual acuity of both the reality and the virtuality, with details in Sect. 3.4. We develop a system calibration algorithm for AR-Loupe, including interactive field-of-vision segmentation and modified stereo-SPAAM to correctly provide overlay in the magnified and non-magnified field-of-vision, with details in Sect. 3.5. The occluded field-of-vision employs a novel method to ensure smooth transition between the magnified and non-magnified field-of-vision, via image warping on the display space, with details in Sect. 3.5.4. Tianyu Song developed the first version of the prototype as a course project for CIS II, under the supervision of me.

3.3 Background and Related Works

3.3.1 Head-Mounted Loupes

A loupe is an optical magnification device to enhance the sight of fine details. Head-mounted loupes are widely used in watchmaking, jewelry industry and healthcare [155]. Dentists wear loupes to observe small cracks in teeth, root canal orifices,

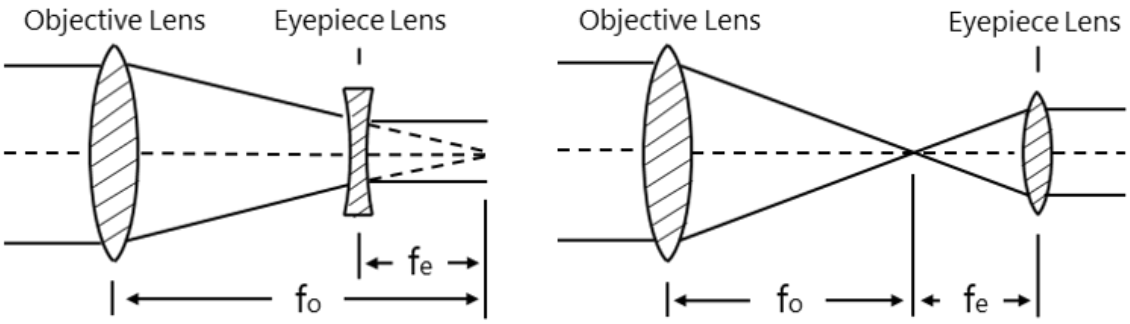


Figure 3.3: Galilean (left) and Keplerian (right) type of loupe

and dental caries [155]. It significantly improves the near visual acuity of dentists [261] and is becoming an accepted norm amongst practitioners [114].

There are two major types of head-mounted loupes: Galilean and Keplerian [193]. The Galilean loupe consists of a convex objective lens and a concave eyepiece lens. The Keplerian loupe uses two convex lenses, and often an additional Schmidt-Pechan prism to invert the image so that it is displayed upright. Fig. 3.3 illustrates a simplified diagram of the optics of Galilean and Keplerian loupes, where f_o and f_e denote the focal length of the objective lens and eyepiece lens, respectively.

The main optical parameters of a loupe system are: magnifying power, barrel length, field of view and depth of field. The magnifying power is the ratio of the sizes of the images formed on the user's retina with and without the loupe [101]. With simplification (thin lens assumption and linear magnification assumption), it can be calculated as $M = f_o/f_e$. As the magnifying power increases, the field that can be viewed decreases. The barrel length of a loupe is the distance separating the objective lens and eyepiece lens, calculated as $f_o + f_e$ for a Keplerian loupe and $f_o - f_e$ for a

Galilean loupe. The working distance of a loupe is the distance where the loupe is focused. The range of the working distance is the depth of field. It is customized for different applications and users' ergonomics.

In general, a Galilean loupe provides less magnification, is lighter and more affordable. In dentistry, a Galilean loupe with magnifying power of 2.5x or 3.5x is most commonly used. A Keplerian loupe offers higher magnification power, but is heavier and more costly, partially due to the additional prism and longer barrel length [114]. A Galilean loupe is used for the AR-Loupe prototype, as shown in the Fig. 3.1.

3.3.2 Zoomable Augmented Reality

We propose the concept of zoomable augmented reality, where the reality viewed by the user can be magnified or minified, and meanwhile, the virtual content remains registered with it. In the literature, similar concepts have been implemented [158, 28, 27, 67, 107] and are reviewed below.

Martin-Gonzalez et al. developed a head-mounted magnification system for surgical application, based on a video see-through head-mounted display (VST-HMD), called Virtual Loupe [158]. A pair of color cameras first capture the surgical site, and then the region-of-interest is digitally magnified to enhance the visualization for the surgeon. However, with Virtual Loupe, the visual acuity of the surgeon does not necessarily increase because the resolution of the picture is fixed at the time of capture. Moreover, the authors did not render additional virtual content for augmented

reality. Huang et al. developed Scope+, which is a stationary AR-enabled microscope based on video see-through technology [107].

Birkfellner et al. developed Varioscope AR, and described the system in a few publications [28, 27, 67]. Varioscope AR is implemented based on a head-mounted surgical microscope (Varioscope) with additional optical combiners and LCD displays. The latest hardware prototype described in based on Varioscope M5 (currently Leica HM500, cost around €$50k$), supports variable zoom and focus, however, it is considerably bulky ($145 \times 70 \times 95\,mm$) [67]. The system requires a surgical workstation and external optical tracking unit to provide properly-registered augmented reality. The calibration of Varioscope AR is performed purely with a camera attached to the eye piece, and the accuracy of calibration is not validated subjectively. Compared to Varioscope AR, AR-Loupe takes advantage of the current generation of OST-HMD. It is more compact, affordable, easier to operate and calibrate, supports AR in both magnified and non-magnified field-of-vision, and is validated both objectively and subjectively.

3.4 Hardware Design of AR-Loupe

We use Magic Leap One as the OST-HMD for AR-Loupe, which has four embedded magnets to position prescription inserts and a forehead pad. We design the loupe attachment with magnet inserts, that is suited to the existing OST-HMD structure, as

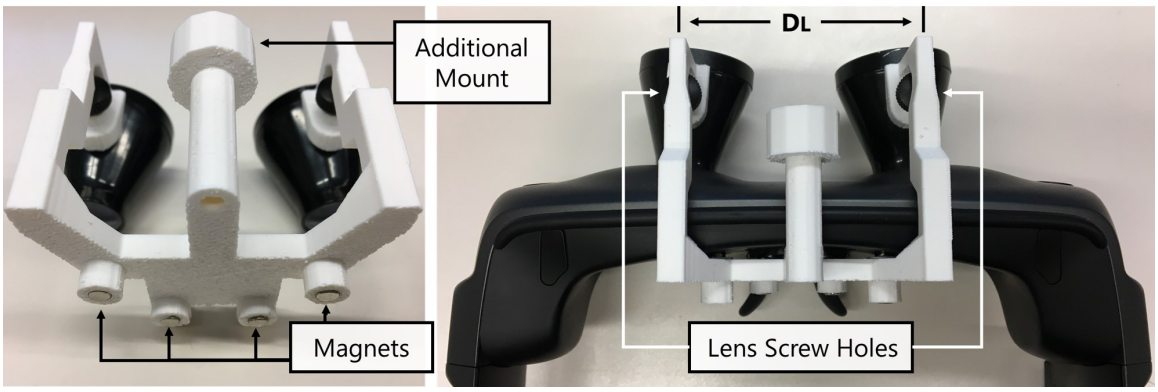


Figure 3.4: Left: the loupe attachment. Right: top view of AR-Loupe. The distance between the two loupe centers is denoted D_L.

shown in Fig. 3.4. Therefore, the attachment can be rigidly affixed to the OST-HMD, to ensure good repeatability.

There is an additional mount in the horizontal center of the attachment. It can be used to position co-axial lighting or other egocentric sensing unit, e.g. depth camera. A Galilean loupe pair is affixed to the attachment using screws. The positions of the screws are fixed on the attachment. The horizontal distance between the binocular loupes is D_L. Once the loupes are inserted, they are oriented inwards with angle α, which is pre-defined by the brackets that hold the loupes. Fig. 3.5 illustrates the geometric parameters of the attachment. The interpupillary distance of the users is denoted as D_E. The working distance (from the eye to the focused object) is D_{EO}, which is composed of D_{EL} (from eye to the loupe) and D_{LO} (from loupe to the focused object).

The attachment can be customized for each user, including the interpupillary distance D_E and preferred working distance D_{EO}. We assume that D_{EL} is fixed,

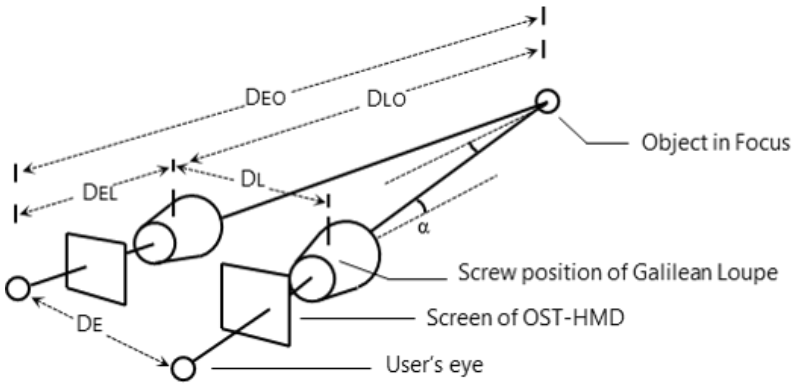


Figure 3.5: Geometric illustration of the components of AR-Loupe

because the distance between the eye and the loupe does not vary significantly, and moreover, it is a relatively small portion of D_{EO}. Therefore, we can calculate the other design parameters as:

$$\alpha = arcsin\Big(\frac{D_E}{2 \cdot D_{EO}}\Big), \quad D_L = D_E \cdot \frac{D_{EO} - D_{EL}}{E_{EO}} \tag{3.1}$$

One more constraint for AR-Loupe is that D_{EO} should be within the working distance of the Galilean loupe. The attachment is prototyped via 3D printing with the specific α and D_L. Then, we install the magnets and fix the binocular Galilean loupe on the attachment using screws. Finally, we assemble the AR-Loupe by magnetically affixing the attachment onto the Magic Leap One.

3.5 Methods

In this section, we model the see-through view of AR-Loupe, develop the user-specific calibration method, and then present the rendering pipeline of AR-Loupe.

3.5.1 Interactive Field-of-Vision Segmentation

The example see-through view with AR-Loupe is shown in Fig. 3.6. A picture of Lena is placed in the background. Inside of the optically magnified area, the face of Lena appears larger than the rest of the picture.

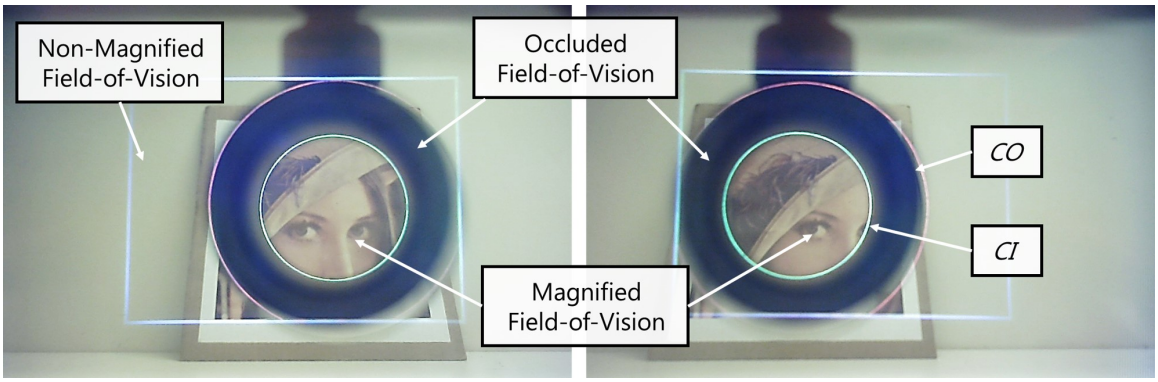


Figure 3.6: Interactive view segmentation of AR-Loupe. The white rectangle is the field-of-view of OST-HMD. The users manipulate the red and the green circles to be aligned with the outer and inner borders of the loupes.

The white rectangle border shows the screen edge of the OST-HMD. In order to provide augmentation across the field-of-view, it is necessary to first accurately segment different areas of the see-through vision. The loupe is seen as a black ring with a handle, in front of the user. To simplify the segmentation procedure, we ignore the handle and model the appearance of the loupe as a ring, comprised of an outer circle CO and an inner circle CI. The user's vision is magnified inside of the inner circle (CI), and remains normal outside of the outer circle (CO). Between CO and CI, the vision is occluded by the structure of the loupe.

With OST-HMD, it is not possible to access the user's retina image. Therefore,

we propose an interactive segmentation procedure, where four circles are rendered on the OST-HMD screen as shown in Fig. 3.6. The red circles are COs and the green circles are CIs. The users manipulate the circles by changing their positions and sizes until they match the borders of the loupes. The parameters that are determined by interactive segmentation are listed in Tab. 3.1.

Table 3.1: Parameters for interactive view segmentation

Eye	Circle	Position (pixel)	Size (pixel)
Left	Inner	$(i^L_{CI},\ j^L_{CI})$	R^L_{CI}
	Outer	$(i^L_{CO},\ j^L_{CO})$	R^L_{CO}
Right	Inner	$(i^R_{CI},\ j^R_{CI})$	R^R_{CI}
	Outer	$(i^R_{CO},\ j^R_{CO})$	R^R_{CO}

After different visual regions are segmented, they are treated separately for calibration and visualization in the following subsections. There may be certain degrees of redundancy in the parameter set. For example, the vertical positions of the loupe in front of both eyes may be the same ($j^L_{CI} = j^R_{CI}$ and $j^L_{CO} = j^R_{CO}$) if we assume that the OST-HMD is balanced on the head. In another example, the size of the circles for left and right may be assumed equivalent ($R^L_{CI} = R^R_{CI}$ and $R^L_{CO} = R^R_{CO}$). Furthermore, it is possible to consider the occluded region as a perfect ring, so that the inner and outer circles are concentric ($(i^L_{CI},\ j^L_{CI}) = (i^L_{CO},\ j^L_{CO})$ and $(i^R_{CI},\ j^R_{CI}) = (i^R_{CO},\ j^R_{CO})$). There is a trade-off between accuracy and convenience for the view segmentation procedure. For critical tasks, more parameters should be calibrated to have more

accuracy.

3.5.2 Modeling AR-Loupe

Within the magnified field-of-vision, the light passes through the loupes as illustrated in Fig. 3.7.

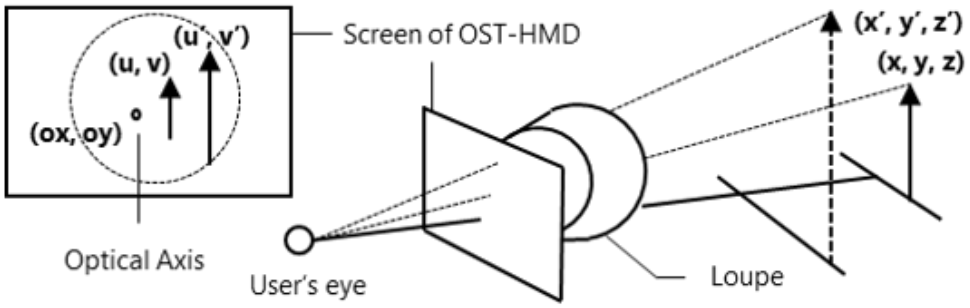


Figure 3.7: The modeling of the projection relationship in the magnified field-of-vision. A 3D point (x, y, z) in the physical world should be projected to (u, v) without the interference of the loupe. But with magnification, it appears at (u', v').

We assume that the user's eye and the screen of the OST-HMD forms a pinhole camera without distortion, which is also a fundamental assumption for SPAAM as discussed in Sect. 2.3.1, and is well-acknowledged in the community [258]. Then, given a 3D point in the camera coordinate system (x, y, z), it should be 'projected' on the screen at pixel location (u, v). The following equation holds:

$$w \cdot \left[\, u \,,\, v \,,\, 1 \right]^T = \left[\, K \,\right] \left[\, R \mid t \,\right] \left[\, x \,,\, y \,,\, z \,,\, 1 \right]^T \tag{3.2}$$

where K is a 3×3 upper triangular matrix depicting the intrinsic parameters of the eye-screen pinhole camera and $R \in SO(3)$. As introduced in Sect. 3.4, based on the assumptions: i) thin lens equation, ii) linear approximation of small angle, and

iii) no distortion caused by lens, the magnifying power is calculated as $M = f_o/f_e$. Therefore, if the optical axis of the loupe is projected to (o_x, o_y), the magnified object should appear at (u', v'), where:

$$\begin{aligned} u' &= M \cdot (u - o_x) + o_x = M \cdot u + (1 - M) \cdot o_x \\ v' &= M \cdot (v - o_y) + o_y = M \cdot v + (1 - M) \cdot o_y \end{aligned} \tag{3.3}$$

Therefore, we can write (u', v') as a linear transformation of (u, v):

$$\left[\, u' \,,\, v' \,,\, 1 \right]^T = \left[\, Q \,\right] \left[\, u \,,\, v \,,\, 1 \right]^T \,, \quad Q = \begin{bmatrix} M & 0 & (1-M) \cdot o_x \\ 0 & M & (1-M) \cdot o_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

Combining Eq. 3.2, Eq. 3.3 and Eq. 3.4, we have:

$$w \cdot \left[\, u' \,,\, v' \,,\, 1 \right]^T = \left[\, P \,\right] \left[\, x \,,\, y \,,\, z \,,\, 1 \right]^T \,, \quad P = \left[\, Q^{-1} \, K \,\right] \left[\, R \mid t \,\right] \tag{3.5}$$

Because Q is an invertible 3×3 upper triangular matrix, Q^{-1} is also a 3×3 upper triangular matrix. Then, $Q^{-1} K$ is another 3×3 upper triangular matrix. Consequently, the relationship between the 3D point (x, y, z) and the magnified view of it on the OST-HMD screen (u', v') is also a projection transformation, denoted P. In other words, the combination of eye, OST-HMD screen and the loupe can be modeled as a pinhole camera. P is an upper triangular matrix and captures the intrinsic parameters of this eye-screen-loupe pinhole camera. The same model also applies to a minifying loupe. With a binocular AR-Loupe, the projection matrix for the left eye is P_L, and that of the right eye is P_R.

3.5.3 Calibrating AR-Loupe

In Sect. 3.5.2, we derive the projection model of the magnified field-of-vision. Therefore, we can apply the traditional display calibration algorithms introduced in Sect. 2.3 to calibrate the optical-zoomed field-of-vision, because the underlying model is identical. For the left eye and right eye, the calibration procedure estimates the 3×4 projection matrix separately: P_L and P_R.

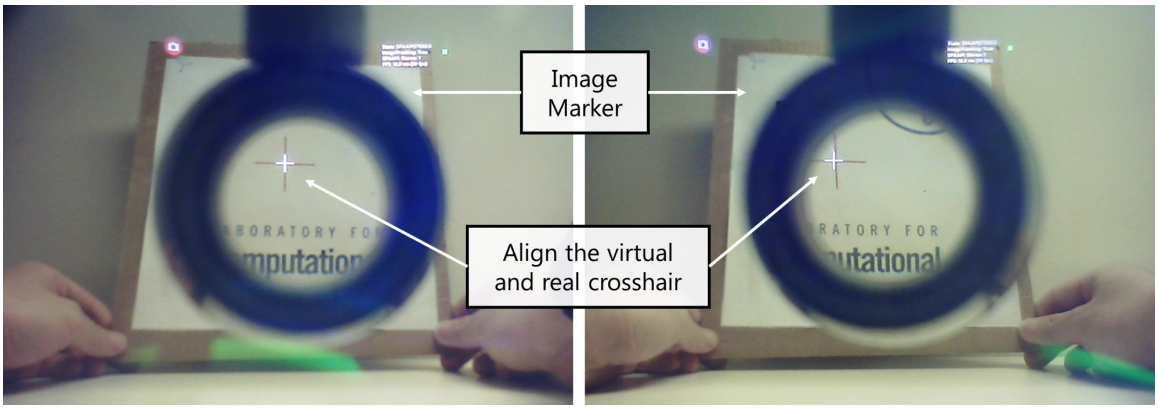


Figure 3.8: The display calibration of AR-Loupe. Two white virtual crosshairs are displayed in the zoomed field-of-vision, seen by the user as a 3D virtual crosshair with depth. The users manually align an image marker (with a crosshair in the center, also shown in Fig. 3.12) with the virtual crosshair.

We choose Stereo-SPAAM as the backbone of our calibration method, with adjustments that are suited for AR-Loupe. Stereo-SPAAM is a classical method for stereoscopic OST-HMD calibration [75]. It offers good accuracy but is not optimized for user-friendliness. We choose Stereo-SPAAM to ensure the accuracy of the display calibration, and at the same time, it allow us to investigate the results of the display calibration which may lead to improvement in calibration efficiency. In Stereo-SPAAM, a target is displayed on both sides of the binocular OST-HMD, so that it

can be seen as a 3D virtual target, as shown in Fig. 3.8. The user holds a paper marker, which is tracked by the OST-HMD, to align the center of the paper marker with the 3D virtual target. Once the user confirms that the target and the marker are aligned via a specific input mechanism (e.g., controller, keyboard, voice), another 3D virtual target is then displayed, and the user repeats the alignment a few times. For the $i-$th alignment, the position of the target on the left and right displays in the pixel coordinate system are recorded: (u_L^i, v_L^i) and (u_R^i, v_R^i). The 3D positions of the center of the paper marker are also recorded: (x^i, y^i, z^i). We then use the Direct Linear Transformation (DLT) algorithm to separately estimate the projection $P_L : \{(x^i, y^i, z^i)\} \rightarrow \{(u_L^i, v_L^i)\}$ and $P_R : \{(x^i, y^i, z^i)\} \rightarrow \{(u_R^i, v_R^i)\}$. More specifically, if we take P_L as an example, the 2D point set $\{(u_L^i, v_L^i),\ i \in [1, N]\}$ and 3D point set $\{(x^i, y^i, z^i),\ i \in [1, N]\}$ are both first normalized into $\{(\bar{u}_L^i, \bar{v}_L^i),\ i \in [1, N]\}$ and $\{(\bar{x}^i, \bar{y}^i, \bar{z}^i),\ i \in [1, N]\}$. Then we construct a $2N \times 12$ matrix B:

$$B_L = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \bar{x}^i & \bar{y}^i & \bar{z}^i & 1 & 0 & 0 & 0 & 0 & -\bar{u}_L^i \bar{x}^i & -\bar{u}_L^i \bar{y}^i & -\bar{u}_L^i \bar{z}^i & -\bar{u}_L^i \\ 0 & 0 & 0 & 0 & \bar{x}^i & \bar{y}^i & \bar{z}^i & 1 & -\bar{u}_L^i \bar{x}^i & -\bar{v}_L^i \bar{y}^i & -\bar{v}_L^i \bar{z}^i & -\bar{v}_L^i \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{3.6}$$

We use DLT to estimate a vector $\vec{p}$ that approximates $B_L \vec{p} = 0$. The same algorithm is repeated for P_R.

We make two adjustments to Stereo-SPAAM for the calibration of AR-Loupe. First, because the users need to align the real object with the virtual target, they should be able to see the real object clearly. It is not an issue for Stereo-SPAAM in a normal setup, however, it needs to be taken into consideration due to the limitation

of working distance introduced by Galilean loupes. We ascertain that our pre-defined 2D targets on screen are fused at a distance that falls into the approximate working distance of AR-Loupe. Secondly, normally, the 2D targets on the left and right screens are rendered at the same height: $v_L^i = v_R^i$. However, through our experiment, we find that the OST-HMD may be a little tilted on the user's head, so if we render the two targets at the same height, the users may not be able to fuse them into one single 3D target. This phenomenon is known as binocular diplopia [185]. We employ an interactive vertical adjustment step, where the user is allowed to manipulate the height of the target rendered on the right screen (v_R^i) until he/she can perform binocular fusion.

3.5.4 Management of Occluded Information

Occlusion refers to the inability to see the physical world. Occlusion occurs with AR-Loupe due to two reasons. First, the structure of the loupe blocks part of the user's field-of-vision. Secondly, optical magnification enlarges the perceived size of real objects that are close to the optical axis, but hides the objects at the periphery. The total occluded area is shown as the crosshatched region in Fig. 3.9.

On the other hand, the screen of OST-HMD is not fully utilized. Both the outside of the outer circle (CO) and the inside of the inner circle (CI) are used to display augmentation. However the area within the circles (dotted region in Fig. 3.9) cannot be used to render augmentation because there is no correspondent real world object.

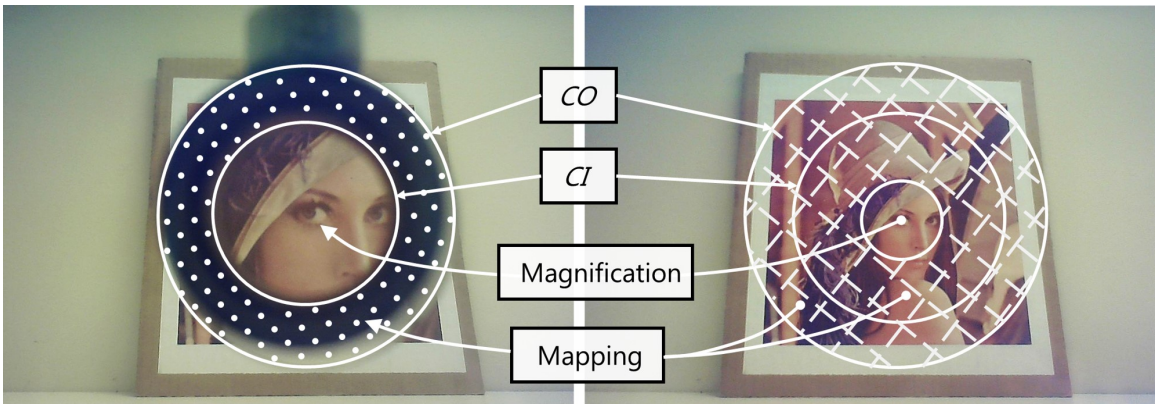


Figure 3.9: There is occlusion with AR-Loupe, due to the hardware structure blocking the light, and the optical lens refracting the light. The crosshatched region shows the occluded area. The dotted region is the portion of the display where we propose to visualize occluded information.

Therefore, we could use this area of the display to visualize the occluded information, including the occluded real object (e.g., the shoulder of Lena) and the augmentation registered with it (e.g., a virtual tattoo on the shoulder). In AR-Loupe, we visualize only the virtual information of the occluded area, by post-rendering image-warping.

3.5.5 Rendering for Zoomable Augmented Reality

The pipeline for the rendering for zoomable augmented reality is illustrated in Fig. 3.10. A graphics scene is set up, containing all virtual objects to be rendered as augmentation. Before the rendering of each frame, the virtual scene is updated with the latest pose of the headset and inputs from other sensing units. The main cameras that represent the normal field-of-view are first rendered (the blue and the light orange regions) into framebuffers (FB_L and FB_R). We set up two additional virtual cameras to render the zoomable views with the calibration parameters. We derive the

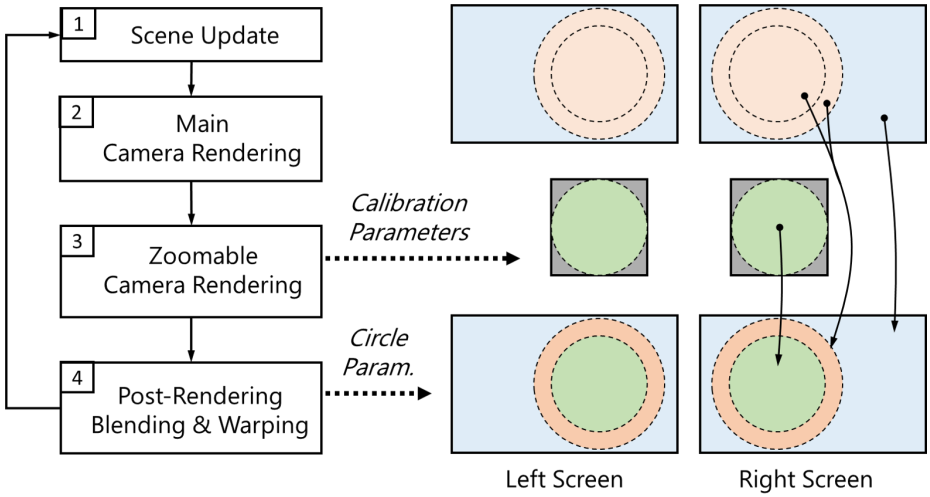


Figure 3.10: The rendering pipeline for AR-Loupe. Post-rendering warping is used to combine the normal and zoomable visualization.

projection matrix (CAM_L and CAM_R) of the virtual cameras from the calibration results (P_L and P_R) presented in Sect. 3.5.3. Similar to [258], the derivation is:

$$
\begin{aligned}
CAM_{\{L,R\}} &= glOrtho(\cdot)\ \left[\, A\,P_{\{L,R\}} + B\ \right] \\
glOrtho(\cdot) &= \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{r-l} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
A &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -(f+n) \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & f \cdot n \\ 0 & 0 & 0 \end{bmatrix}
\end{aligned}
\tag{3.7}
$$

where r, l, t, b, f, n are parameters defining the left, right, top, bottom, far and near culling plane. Note that $P_{\{L,R\}}$ is 3×4 matrix while $CAM_{\{L,R\}}$ is 4×4 matrix. The zoomable cameras also render to framebuffers, e.g. FB_{ZL} and FB_{ZR}. At the post-rendering stage, blending and warping is performed using a fragment shader. The inputs to the fragment shader are the four framebuffers (FB_L, FB_R, FB_{ZL} and FB_{ZR}) and the parameters of view segmentation in Sect. 3.1. The green regions in

FB_{ZL} and FB_{ZR} replace the content in the inner circles of FB_L and FB_R. The corners of FB_{ZL} and FB_{ZR} (gray region) are discarded. The occluded area in FB_L and FB_R (light orange region) are compressed into the dark orange region to provide evidence of virtual content within the border of the loupe frame.

Figure 3.11: Example visualization with AR-Loupe. The center and the border of the image marker are augmented (in white). The augmentation is registered across the entire field-of-view after calibration, including both the magnified area and the normal see-through area of the screen. The border of the loupe is used to provide occluded information.

3.6 Implementation

We choose Magic Leap One as the headset for AR-Loupe, which offers binocular optical see-through displays and relatively wide field-of-view (50° horizontal). The fact that the computation unit is separated makes the head-mounted piece lighter and less bulky. The binocular loupe used in our prototype is AZDENT® Dental Magnifier, which costs around $35. It is a Galilean loupe, with a magnification power

of $3.5x$ and with a working distance of $280mm \sim 380mm$. The attachment mounting piece is 3D printed with custom interpupillary distance and working distance, as discussed in Sect. 3.4. In total, the hardware of AR-Loupe costs about $\$2.5k$.

The software of AR-Loupe is implemented based on Unity. Magic Leap One runs on Lumin OS, which is a customized Android OS. Magic Leap offers a development kit (Lumin SDK) for Unity, which includes the interface with its 6-DOF controller, feature-based image tracking, pose estimation of the headset, eye tracking and single-pass stereo rendering. We designed an image for tracking and calibration, shown in Fig. 3.12. It has a crosshair at the image center, for the ease of user alignment. We build a dynamic link library (C++) for the zoomable augmented reality calibration based on Eigen. We use Unity Native Plugins to interface the dynamic link library at runtime for best efficiency. The post-rendering fragment shader is written in Unity ShaderLab. AR-Loupe constantly achieves $60\,Hz$ rendering framerate.

3.7 Verification

In Sect. 3.5.3, we successfully model the combination of eye, OST-HMD screen and the Galilean loupe as a pinhole camera, which can be characterized by a 3×4 projection matrix. In this subsection, we objectively verify our model of zoomable augmented reality, using a pair of eye-simulating cameras placed behind the AR-Loupe. The setup is shown in Fig. 3.12. The two color cameras are separated by

66.8mm, which is the targeted interpupillary distance of the attachment piece.

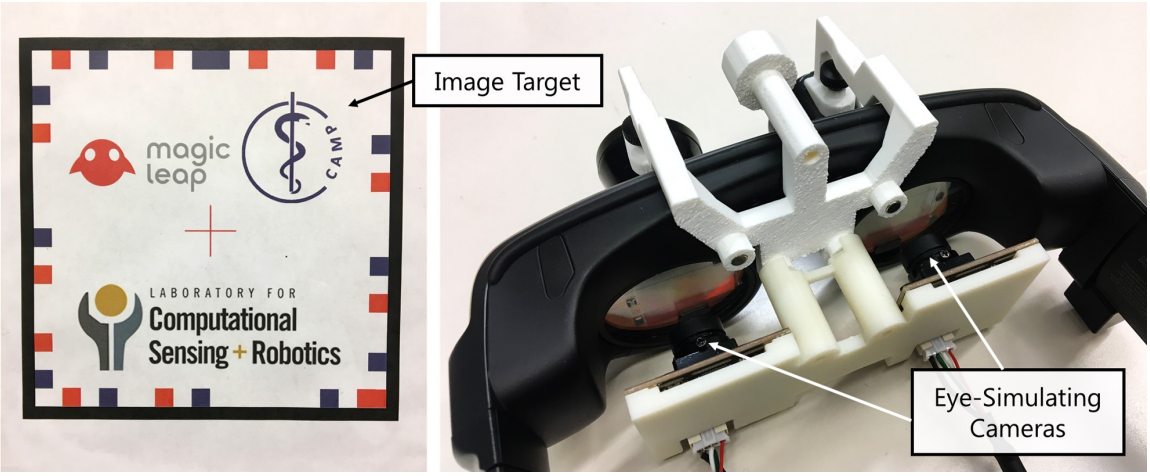


Figure 3.12: Left: the image marker that can be tracked by Magic Leap One, with a crosshair in the center to be aligned with a virtual target. Right: the setup for objective verification of the calibration methods.

During the verification, we manually position the image marker in front of the AR-Loupe so that the center crosshair is seen through the magnified field-of-vision, as seen in Fig. 3.13 (but without displaying the white virtual crosshair). We develop an application based on computer vision to find out the location of the real crosshair in the captured images. First, the non-magnified field-of-view is masked out. The image mask is manually created, and remains the same across all frames because the eye-simulating cameras are fixed w.r.t. the loupes. Secondly, the edges on the image are enhanced, and followed by Hough Transform-based line detection. Then, we select the first two line candidates, and compute their intersection, noted as (u_i, v_i). We also retrieve the tracking data at the time of capture, i.e., the position of the crosshair is (x_i, y_i, z_i). We manually moved the image marker with varying distances

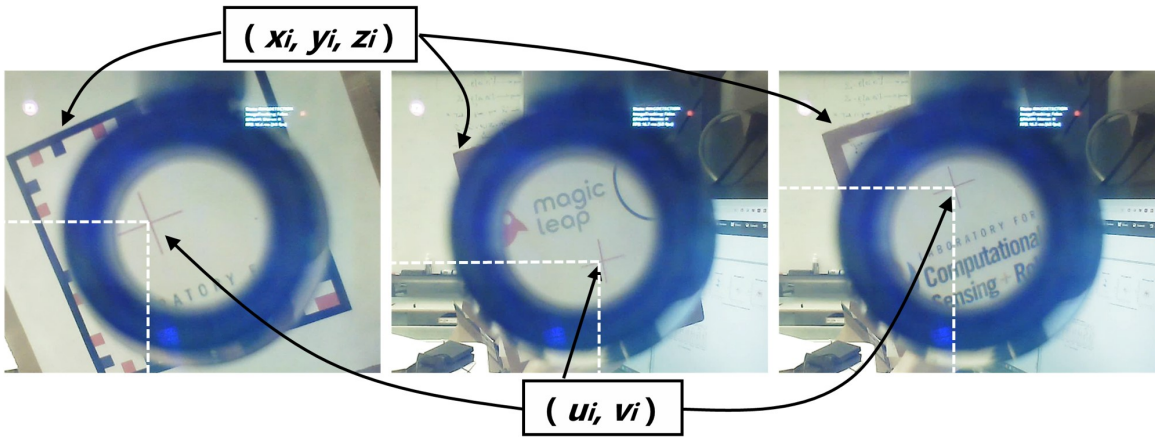


Figure 3.13: During the objective verification, the AR-Loupe tracks the 3D position of the image target as (x_i, y_i, z_i), and a computer-vision-based algorithm estimates the 2D position of the crosshair on the captured image as (u_i, v_i).

to AR-Loupe, and collected 64 instances of (u_i, v_i) and (x_i, y_i, z_i).

Note that (u_i, v_i) is the pixel location of the crosshair on the eye-simulating camera image, not the pixel location of the OST-HMD. However, based on the assumption that the image plane of the eye-simulating camera is parallel to the screen of the OST-HMD, it is sufficient to verify that a projection transformation is able to capture the mapping: $f(\cdot) : \{(x_i, y_i, z_i) \rightarrow (u_i, v_i),\ \forall i \in [\,1, 64\,]\}$.

We run the DLT algorithm to find the mapping $f(\cdot)$. We calculate the average reprojection error as the residue of the projection mapping. The average reprojection error is 1.56 pixel, which corresponds to about 0.2° visual angle. Therefore, we verified that our model of zoomable augmented reality can sufficiently capture the optical system. The remaining reprojection error can be due to the distortion caused by the refraction of the lens.

3.8 Experiments

We designed and conducted a two-phase multi-user study to evaluate the accuracy and usability of AR-Loupe. We hypothesize that AR-Loupe can help the users to perform accuracy-demanding tasks, with increased visual acuity and augmentation with finer details. Therefore we set up a basic AR guidance application to evaluate our hypothesis. In the application, we use AR-Loupe to provide augmentation at certain known positions with respect to the image marker. 9 virtual crosshairs are displayed on the same plane of the image marker, seen in Fig. 3.14. The users are asked to mark down the centers of the crosshairs on the paper marker.

Figure 3.14: A simplified AR guidance task with AR-Loupe (Left), compared with normal AR guidance (Right). The image target is augmented with 9 crosshairs at known positions.

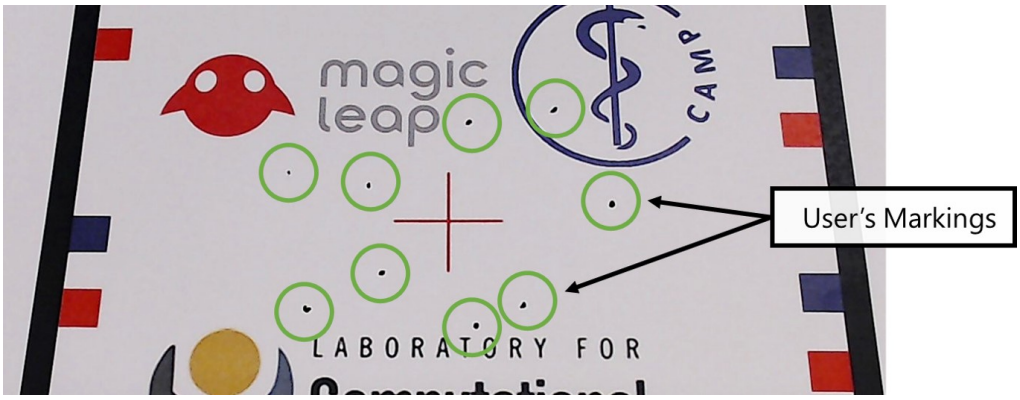


Figure 3.15: The users marked down the positions of the augmentations on the marker image, under the guidance of AR-Loupe or normal AR corresponding to Fig. 3.14.

3.8.1 User Demographics

We recruited 9 users (7 male, 2 female, average age: 28.1) from the Department of Computer Science at Johns Hopkins University to participate in the user study, under IRB approval (Homewood Internal Review Board No. 00007467). Each user filled in a pre-experiment survey including basic user information, including whether he/she has uncorrected vision, and the interpupillary distance (IPD). One user was excluded due to uncorrected vision (astigmatism), which caused him to be unable to binocular fuse the virtual content.

For the users who are not aware of their IPD value, we use the Microsoft HoloLens Calibration app to measure it. We manufactured four sets of 3D-printed attachments, with working distance calibrated to $35mm$, and IPD customized to $60mm$, $62.8mm$, $65.6mm$ and $68.4mm$ respectively. Once the user IPD is obtained, we choose the most suitable attachment and assemble AR-Loupe. We confirmed with the users that they

could clearly see magnified objects within the working distance.

3.8.2 First Phase - Comparison Study

3.8.2.1 Guidance with AR-Loupe

We introduced the calibration and evaluation procedure of AR-Loupe to the users, as listed below. The users only need to interact with the controller to complete the calibration steps and evaluation.

1. AR-Loupe displays a circle on the left screen, and the user fits the inner circle (position (i^L_{CI}, j^L_{CI}) and radius R^L_{CI}) to the border of the magnified area, for the left eye.
2. AR-Loupe displays a circle on the right screen, and the user fits the inner circle (position (i^R_{CI}, j^R_{CI}) only, radius equal to R^L_{CI}) to the border of the magnified area, for the right eye (The calibration procedure is simplified, for example, the segmentation of outer circles ((i^L_{CO}, j^L_{CO}), R^L_{CO}, (i^R_{CO}, j^R_{CO}), R^R_{CO}) is skipped, because these parameters are not relevant for our evaluation task).
3. AR-Loupe displays two crosshairs on the left and right screen within the inner circles, and the user can move the vertical position of the crosshair on the right screen so that he/she can see only one crosshair most comfortably (binocular fusion).
4. The user then holds the image marker (Fig. 3.12) to align the center crosshair

of the image marker with the virtual crosshair on Magic Leap One. The user uses the controller to confirm the alignment.

5. Steps 3 and 4 are repeated 16 times, with the virtual crosshair at different locations. At each time of alignment confirmation, AR-Loupe records the current 2D positions of the crosshair, (u_L^i, v_L^i) and (u_R^i, v_R^i), and the current 3D position of the image marker (x^i, y^i, z^i).
6. The projection matrices (P_L and P_R) for zoomable augmented reality are calculated after 16 alignments are done. The projection matrices are then plugged into the rendering pipeline.
7. 9 virtual crosshairs with known positions on the image marker plane are displayed with AR-Loupe (Fig. 3.14). The users mark down the centers of the crosshairs on the image marker using a pen (Fig. 3.15).

After the users familiarized themselves with the AR-Loupe, they calibrated the system, and completed the guidance task with augmentation through the calibrated AR-Loupe. The paper marker with pen marking, and the Unity application log file are saved for further processing.

3.8.2.2 Guidance with Normal AR

The users also completed the same guidance task with the normal AR of Magic Leap One, without attachment or the loupe, as shown in the right part of Fig. 3.14. Same as step 9 of Sect. 3.8.2.1, 9 virtual crosshairs were displayed on the image marker

plane, and the user marked down the positions of them on the paper marker.

The application of normal AR guidance is also developed using Unity, based on the Magic Leap ImageTracking Example. Note that additional user calibration is not needed for this application, once the Magic Leap Visual Calibration is done (part of pre-experiment familiarization). The Magic Leap Visual Calibration is an eye tracker calibration. The display calibration of Magic Leap takes advantage of its embedded eye tracking sensors. For the baseline evaluation, the paper with marked crosshair centers and the Unity application log file are also saved for further processing.

3.8.2.3 Subjective Evaluation

After the guidance tasks with AR-Loupe and normal AR setups, the users completed the post-experiment survey about their experience of the experiment, which includes the following two parts:

1. For the **calibration** of AR-Loupe, the users reported the task load of the procedure. The standard NASA Task Load Index (NASA-TLX) is used [98].
2. For the **evaluation** task, the users rated their subjective feeling about the *Outcome*, *Speed*, *Confidence*, *Satisfaction*, *Fatigue* (Fatigue ratings: 0 (very tired) to 5 (very relaxed)), *Interest*, and *Clarity*, for AR-Loupe and normal AR separately.

The subjective rating for the evaluation task is scaled from 0 to 5, with 5 representing the best subjective feeling. For *Fatigue*, 5 means that the user feels very

relaxed, and 0 means that the task causes much fatigue. *Clarity* refers to the clarity of the visualization of the virtual crosshair, which is a subjective measure of the visual acuity.

3.8.3 Second Phase - Repeatability Study

In the first phase of the study, the users are relatively inexperienced with AR-Loupe. It may take more time or effort for them to complete the calibration. We invited the subjects to participate in the second phase of the study a few days later. We evaluate whether their performance and the effort level have changed after they have gained some experiences with AR-Loupe.

In the second phase of the study, the users calibrate the AR-Loupe and perform the evaluation task under AR-Loupe guidance two times (same as Sect. 3.8.2.1). They are provided with the same attachment for the AR-Loupe assembly. After that, they fill in the post-experiment survey again (NASA-TLX for calibration, and subjective ratings for evaluation). The papers with ink mark and application log files are saved.

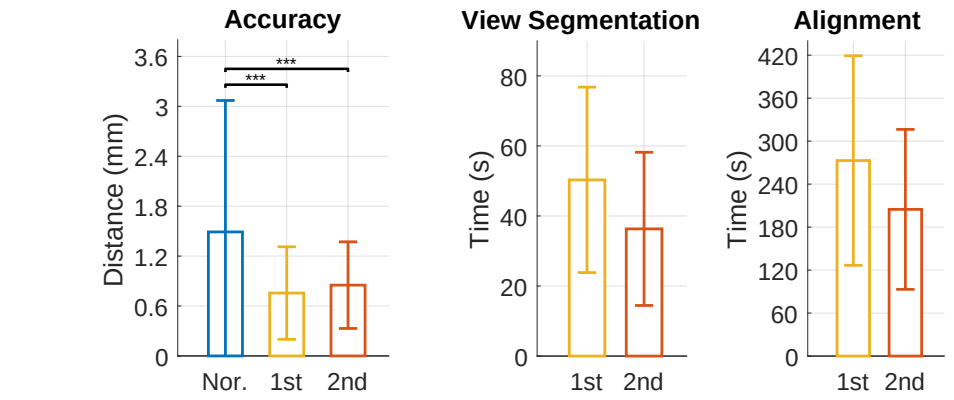
3.8.4 Data Extraction

After both phases of the study, we obtained 4 Unity application log files, and 4 paper markers with marked center points (Normal AR $\times 1$, first-phase AR-Loupe $\times 1$, second-phase AR-Loupe $\times 2$) for each subject. We extract useful data from them and

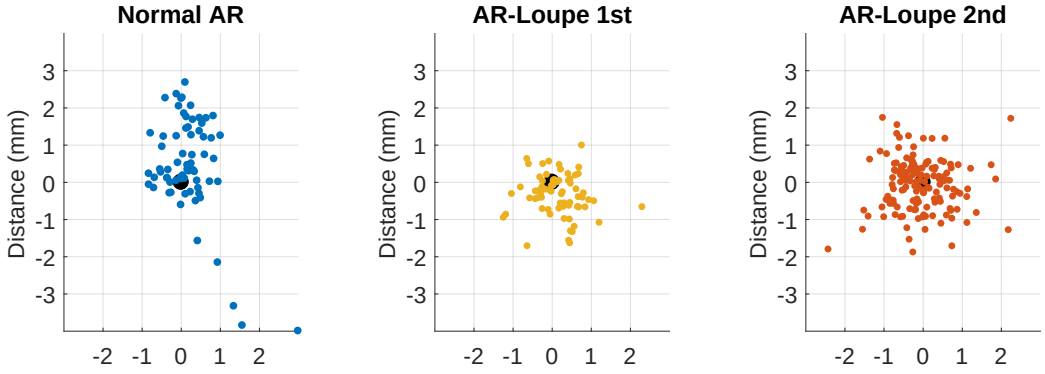
develop a few metrics for evaluation.

The application log files include the timestamps for each user interaction, and the parameters of the calibration procedure. We extract the total time that the user spends on view segmentation T_{SEG} (calibrating the inner circles for left and right eyes) and on Stereo-SPAAM T_A (making alignments between the real and virtual crosshair). We also extract the position and size of the inner circles during the view segmentation. View segmentation has been done 3 times for each user, which allows us to evaluate the repeatability of these parameters.

With the paper markers, we evaluate the guidance accuracy. We first use a camera to capture the pieces of paper markers, then we use computer vision to measure the distance between the user's marking and the ground truth. These pictures are mapped to the original marker image using homography. Then we detect the 2D positions of the markings on the warped image space as $q_i, \forall i \in [1, 9]$ (center of image as $(0, 0)$). At the same time, we know the ground truth positions of the guidance, as $p_i, \forall i \in [1, 9]$ in the image space. The error is defined as the Euclidean distance between p_i and q_i as $E_i = \|p_i - q_i\|$. There are 9 sample points for each user in each evaluation. Combining both phases of the study, there are 72 ($8\ users \times 9\ points$) sample points for normal augmented reality, 72 points for AR-Loupe visualization in the first phase, and 144 points for AR-Loupe visualization in the second phase.



(a) E_{Normal}, E_1 & E_2 **(b)** Temporal performance: T_{SEG} & T_A



(c) The distribution of E_{Normal}, E_1 and E_2. There are 4 points outside of the figure scope for E_{Normal}.

Figure 3.16: Results of the evaluation of AR-Loupe: (a) shows the accuracy of the user's performance under AR-Loupe guidance, compared with normal AR setup; (b) shows the average time for users to complete view segmentation and alignment during the calibration of AR-Loupe; (c) shows the error distribution for each experiment condition. (Better seen in color)

3.9 Results and Discussion

3.9.1 Accuracy

The most important metric of using AR for the guidance task is the guidance accuracy. The overall accuracy is shown in Fig. 3.16a and Tab. 3.2. In the normal

Table 3.2: AR-Loupe evaluation results corresponding to Fig. 3.16. Data is presented as mean ± std. The smallest number for each experiment setup is highlighted using bold font.

Metric	Normal	AR-Loupe 1st	AR-Loupe 2nd	AR-Loupe All
Error E (mm)	1.49 ± 1.58	$\mathbf{0.76} \pm 0.56$	$0.85 \pm \mathbf{0.52}$	0.82 ± 0.53
Time T_{SEG} (s)	N/A	50.30 ± 26.48	$\mathbf{36.32 \pm 21.87}$	40.98 ± 23.89
Time T_A (s)	N/A	272.92 ± 146.24	$\mathbf{204.76 \pm 111.65}$	227.48 ± 125.36

AR situation, the average accuracy (The format is *mean value* ± *standard deviation*) is $E_{Normal} = 1.49 \pm 1.58mm$. In the first phase of using AR-Loupe, the average accuracy is $E_1 = 0.76 \pm 0.56mm$. In the second phase of the study, the accuracy is evaluated as $E_2 = 0.85 \pm 0.52mm$. Note that E_{Normal}, E_1 and E_2 consist of 72, 72, and 144 samples, respectively.

We first use Shapiro-Wilk parametric hypothesis test to determine whether the null hypothesis of composite normality is a reasonable assumption regarding the distribution of the error value [230]. The p-values of the Shapiro-Wilk tests are: $p = 4.67 \times 10^{-9}$ for normal setup, 1.52×10^{-5} for the first phase of AR-Loupe, 2.97×10^{-7} for the second phase of AR-Loupe, and 1.74×10^{-9} for both phases of AR-Loupe combined. The Shapiro-Wilk test reveals that it is sufficient to assume that the error values follow normal distribution, which will allow us to further statistically compare the mean error value for different experiment conditions.

With an unpaired t-test, it is determined that AR-Loupe significantly improves the guidance accuracy $p(E_{Normal},\ E_1) = 2.83 \times 10^{-4}$ and $p(E_{Normal},\ E_2) = 1.63 \times 10^{-5}$.

If we combine the samples with AR-Loupe (E_1 and E_2), the overall accuracy E_{Loupe} is $0.82 \pm 0.53mm$, which is again significantly smaller than normal AR guidance $p(E_{Normal}, E_{Loupe}) = 1.38 \times 10^{-7}$. AR-Loupe achieves, in average, sub-millimeter accuracy for the guidance task. There is no significant difference in terms of accuracy for the first and second evaluations. The accuracy improvement is repeatable.

The detailed error distribution is shown in Fig. 3.16c. Each dot represents one data point of evaluation. For the baseline situation with normal augmented reality, there are a lot more 'outliers'. 4 points are not in the range of this error map, and therefore are not visualized. With AR-Loupe, the points are relatively gathered at the center. Note that E_2 contains $2\times$ points than E_{Normal} and E_1.

3.9.2 Temporal Performance

The time taken to calibrate AR-Loupe mainly consists of the time for view segmentation (T_{SEG}) and alignment (T_A), which are shown in Fig. 3.16b and Tab. 3.2. In the first phase of evaluation, users in average take $50.30 \pm 26.48\,s$ to perform the view segmentation (fitting the inner circle for both eyes), and $272.92 \pm 146.24\,s$ to make 16 Stereo-SPAAM alignments. During the second phase, users in average take $36.32 \pm 21.87\,s$ to segment the view and $204.76 \pm 111.65\,s$ to make alignments.

Similar to Sect. 3.9.1, we first conduct Shapiro-Wilk test to determine whether T_{SEG} follows normal distribution. The p-values are: $p = 0.45$ for the first phase, $p = 4.52 \times 10^{-4}$ for the second phase. We can infer from the statistical analysis that

T_{SEG} for the first phase of the study includes more outliers that drive the statistics from a normal distribution, but when the user gained more experience in the second phase of the study, T_{SEG} is more consistent.

We apply one-sided Mann–Whitney U test [156] instead of t-test, which is a non-parametric method to test the alternative hypothesis that the mean of T_{SEG} in the second phase is greater than that of the first phase. The p-value is determined to be $p = 0.059$.

We apply the same statistical analysis to the time for alignment (T_A). Shapiro-Wilk test revealed that the T_A is not a normal distribution for the first phase $p = 0.95$, but is likely a normal distribution for the second phase of the experiment $p = 5.13 \times 10^{-3}$. The one-sided Mann–Whitney U test revealed that the two distributions are not significantly different $p = 0.13$.

Therefore, we can conclude that both T_{SEG} and T_A are decreased when the users have gained some experience, but the improvements are not statistically significant with one-sided Mann–Whitney U test.

Combining all the temporal information, the average calibration time of AR-Loupe is $268.46\,s$ (less than 5 minutes).

3.9.3 Subjective Ratings

We use NASA-TLX to subjectively evaluate the task load of the calibration for AR-Loupe, and the results are shown in Fig. 3.17 and Tab. 3.3. The yellow and

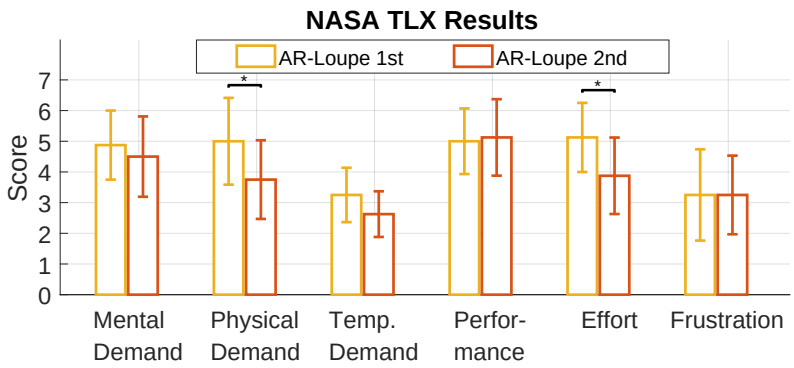


Figure 3.17: Subjective task load rating for the calibration ($N = 8$)

Table 3.3: Subjective task load rating for the calibration corresponding to Fig. 3.17. Data is presented as mean ± std. The better number for each experiment setup is highlighted using bold font. ($N = 8$)

Metric	AR-Loupe 1st	AR-Loupe 2nd	p-value
Mental Demand	$4.88 \pm \mathbf{1.13}$	$\mathbf{4.50} \pm 1.31$	0.34
Physical Demand	5.00 ± 1.41	$\mathbf{3.75 \pm 1.28}$	$\mathbf{3.78 \times 10^{-2}}$
Temporal Demand	3.25 ± 0.89	$\mathbf{2.63 \pm 0.74}$	0.12
Performance	$5.00 \pm \mathbf{1.07}$	$\mathbf{5.13} \pm 1.25$	0.47
Effort Level	$5.13 \pm \mathbf{1.23}$	$\mathbf{3.88} \pm 1.25$	$\mathbf{3.82 \times 10^{-2}}$
Frustration	$\mathbf{3.25} \pm 1.49$	$\mathbf{3.25 \pm 1.28}$	0.54

the red bar captures the task load for the first and second phase of the experiment, respectively.

Since the number of samples is small, we do not assume normal distribution of the data. We apply one-sided Wilcoxon signed rank test [288] to test the null hypothesis that one distribution has larger mean than the other. The results are shown in the fourth column of Tab. 3.3. The physical demand and the effort level have been significantly decreased ($p = 3.78 \times 10^{-2}$ and $p = 3.82 \times 10^{-2}$). Overall, the statistics

reveal that the calibration is becoming easier in the second-phase study, when the users already have experience with calibrating AR-Loupe.

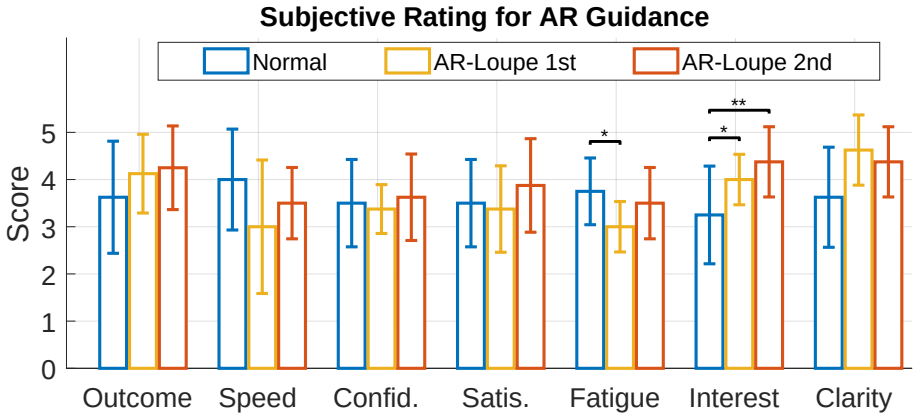


Figure 3.18: Subjective questionnaire for the evaluation task ($N = 8$)

Table 3.4: Subjective questionnaire for the evaluation task corresponding to Fig. 3.17 ($N = 8$). The two phases of the data of AR-Loupe are combined. Data is presented as mean $\pm$ std. The highest mean value and smallest standard deviation for each experiment setup are highlighted using bold font.

Metric	Normal	AR-Loupe	p-value
Outcome	3.63 ± 1.19	$\mathbf{4.18 \pm 0.83}$	0.12
Speed	$\mathbf{4.00 \pm 1.07}$	3.25 ± 1.12	5.83×10^{-2}
Confidence	$\mathbf{3.50} \pm 0.93$	$\mathbf{3.50 \pm 0.73}$	0.35
Satisfaction	$3.50 \pm \mathbf{0.93}$	$\mathbf{3.63} \pm 0.96$	0.31
Less Fatigue	$\mathbf{3.75} \pm 0.71$	$3.25 \pm \mathbf{0.68}$	7.45×10^{-2}
Interest	3.25 ± 1.04	$\mathbf{4.19 \pm 0.66}$	$\mathbf{1.32 \times 10^{-2}}$
Clarity	3.63 ± 1.06	$\mathbf{4.50 \pm 0.73}$	$\mathbf{2.15 \times 10^{-2}}$

We also developed the $0 \sim 5$ point-scaled subjective questionnaire about the evaluation task as described in Sect. 3.8.2.3. The results are shown in Fig. 3.18. The blue columns represent the subjective ratings for the normal AR setup, which does

not require any calibration other than the eye tracking calibration. With normal AR, the field-of-view is not magnified and not occluded as seen in Fig. 3.14, therefore, it is easier and quicker to localize all the crosshair augmentations. The subjective ratings confirmed that the users believed that they are faster to complete the experiment following the guidance, and it caused less fatigue.

We combine the subjective ratings for the two phases of AR-Loupe evaluation, and the results are shown in Tab. 3.4. We use one-sided Mann-Whitney U test to evaluate whether the difference in mean value of the metrics between normal condition and AR-Loupe condition is significant. For the outcome, speed, confidence, satisfaction, and fatigue level, there is no significant difference determined between the two conditions. The users did find that using AR-Loupe to complete the task is more interesting ($p = 1.32 \times 10^{-2}$). And more importantly, they confirmed that the clarity of visualization is significantly improved using AR-Loupe ($p = 2.15 \times 10^{-2}$), which validated our hypothesis that AR-Loupe is able to improve the visual acuity.

Looking at the comparison between the first and second phase of using AR-Loupe, we found that the users thought they were slightly faster (17%), confident (7%), and more satisfied (15%) with their performance in the second time under AR-Loupe guidance. However, none of the subjective metric shows statistical significance.

3.9.4 Repeatability of View Segmentation

In this section, we look at the repeatability of the calibration data. Specifically, we compare the position of the inner circles ((i^L_{CI}, j^L_{CI}) and (i^R_{CI}, i^R_{CI})) and the radius of them (R_{CI}) between the user's separate calibration trials of AR-Loupe. In total, each user calibrated the AR-Loupe three times (first-phase $\times 1$, second-phase $\times 2$). Based on these data, we propose to evaluate the **short-term repeatability** and **long-term repeatablity** of the user's calibration.

If we denote the evaluated variable as Δ, then we have three measurements for each user: Δ_1 is collected in the first phase of the study, while Δ_2 and Δ_3 are collected in the second phase of study. We define the short-term and long-term repeatability as:

$$\phi_{Short}(\Delta) = \frac{1}{N}\sum \left\| \frac{\Delta_3 - \Delta_2}{\Delta_2} \right\|, \quad \phi_{Long}(\Delta) = \frac{1}{N}\sum \left\| \frac{\Delta_2 - \Delta_1}{\Delta_1} \right\| \tag{3.8}$$

where N is equal to the number of users. Here, the evaluated parameter, Δ, can be $i^L_{CI}, j^L_{CI}, i^R_{CI}, i^R_{CI}$ and R_{CI}. The sampling time between Δ_1 and Δ_2 is a few days, and that between Δ_2 and Δ_3 is a few minutes. We heuristically define the repeatability as the average absolute percentage change of a parameter. The results are shown in Tab. 3.5.

Table 3.5: Short-term and long-term repeatability for view segmentation

Parameter	i^L_{CI}	j^L_{CI}	i^R_{CI}	j^R_{CI}	R_{CI}
Short-term $\phi_{Short}(\cdot)$	0.98%	10.77%	2.66%	7.73%	5.37%
Long-term $\phi_{Long}(\cdot)$	4.87%	31.69%	4.87%	25.00%	4.40%

As can be observed from the data, short-term repeatability of the view segmentation is relatively small, because the users may put the AR-Loupe in a similar pose on the head for the two calibration tasks. The Y-axis value of the circle, which represents the height of the magnified area on the user's head, has changed to a larger extent, compared to the horizontal axis because the AR-Loupe is less likely to shift horizontally. The long-term repeatability measured how the parameter has changed over a few days. Most of the parameters have changed more in the long-term than in the short-term, which is aligned with our expectation. The shifts in the vertical axis are larger than the horizontal axis for long-term as well.

In the literature, researchers have been concerned with the repeatability of the display calibration of OST-HMDs. Genc et al. proposed Easy-SPAAM to reuse previous calibration results with a few additional alignments [77]. In the current calibration methods for AR-Loupe, we do not include the information of historical calibration, i.e., every calibration is treated as a complete new procedure.

3.10 Limitations and Future Work

We built the hardware prototype AR-Loupe which is based on a currently available OST-HMD, Magic Leap One. While it offers great convenience for hardware customization and software development, some of its inherent features introduced error to our calibration. First, the display of Magic Leap One has two accommodation

planes. It automatically detects the user's focus distance and chooses the closest accommodation plane for rendering. However, the switch between accommodation planes causes some noticeable drift of the image. Therefore, part of the alignment data may include the 'drift' as well. Second, a few users reported that the augmentation with AR-Loupe is jittery. After diving into the issue, we realized that it is due to the instability of the self-localization of the device. Despite the two issues, we still choose Magic Leap One over some other types of OST-HMD, e.g. Microsoft HoloLens 1st generation, because it has larger field-of-view and shorter distance from the eye to the loupe (D_{EL} in Fig. 3.5) when a loupe is attached to the exterior of the display.

A complete calibration is required each time before using AR-Loupe, without taking advantage of the data from the user's previous calibration or other users' calibration data. It results in an average $4.47\,min$ calibration time. However, as seen from the repeatability evaluation in Sect. 3.9.4, there is definite correlation between the same user's calibration. Potentially, with prior information, the calibration time could be reduced, hence, the user experience could be improved.

As introduced in Sect. 3.8.2.2, the alignment in the non-magnified field-of-vision is provided by Magic Leap One, taking advantage of eye tracking capability. It requires an eye tracking calibration. However, the Magic Leap One SDK does not provide direct access to the eye position data. Once the access is granted, it is useful to exploit alternative ways for the display calibration of the magnified field-of-vision, with the explicit position of the eye, the loupe w.r.t. the display. Potentially, the

projection matrix of the eye-screen-loupe virtual camera can be directly estimated. It will significantly improve the user experience as well.

It is very important to discuss the potential use case with clinicians that frequently use loupes, and identify the system requirements. After that, application-specific experiments should be carried out to evaluate AR-Loupe in a clinical context, for example, in simulated root canal treatment. We also intend to integrate loupes with other OST-HMDs that will enter the market soon, e.g. Microsoft HoloLens 2.

3.11 Conclusion

In this chapter, we introduced the concept of zoomable augmented reality, where the user's field-of-vision can be magnified or minified, and the virtual content appears registered with the real objects across the user's field-of-vision. We developed a zoomable AR prototype, AR-Loupe, integrating Magic Leap One and a binocular Galilean magnifying loupe, with customized attachments catering to the user's interpupillary distance and targeted working distance. We successfully modeled the combination of the user's eye, screen of the OST-HMD, and the optical loupe as a pinhole camera, with some simplifications and assumption. The eye-screen-loupe pinhole-camera model is also verified to be sufficient. In order to calibrate AR-Loupe for precise overlay, the users first segment the field-of-vision interactively, and then perform an adapted Stereo-SPAAM display calibration. We conducted a two-phase multi-

user study to evaluate AR-Loupe in a simple AR guidance task. With AR-Loupe, the users are able to achieve an average sub-millimeter accuracy of $0.82mm$, which is significantly smaller compared to the normal AR guidance accuracy of $1.49mm$, $p = 1.38 \times 10^{-7}$. The mean calibration time is $268.46\,s$. The subjective ratings revealed that the task load for calibration remains high, but is lowered in almost all aspects in the second-phase of the study when the users are more familiar with the system. The users also reported significantly improved visual acuity of the augmentation under the guidance of AR-Loupe ($p = 2.15 \times 10^{-2}$). The repeatability study results suggested that a few calibration parameters have not changed much between different trials, which could be used to potentially ease the calibration procedure.

3.12 Closing Remarks

We integrate an optical magnifier with an OST-HMD to increase the visual acuity of both the reality and the virtuality. However, building optical magnification within the optical system of the OST-HMD would make the system much more compact and easy to use. It is a huge opportunity for the market. I think in the near future, there will be dedicated hardware systems that enable zoomable augmented reality for specific use case, e.g., dental procedures.

3.13 Published Work

Material from this chapter is currently in review for the IEEE Transactions on Visualization and Computer Graphics.

Chapter 4

A "Virtual Monitor" on OST-HMD

This chapter presents the research contributions related to developing a mixed reality visualization technique for medical images, the "Virtual Monitor", based on OST-HMDs to aid image-guided surgery (IGS). It is a simple yet effective concept that has the potential to be deployed in the operating room. We implemented the virtual monitor and evaluated it in percutaneous spine phantom-based studies. We further discuss the evaluation criteria of current OST-HMDs for their suitability for "Virtual Monitor" visualization.

4.1 Introduction

Every day, countless image-guided surgeries (IGS) are conducted by a diverse set of clinicians across many disciplines. From procedures performed by ultrasound tech-

nicians [17, 222, 194], to orthopaedic surgeons [245, 53], to interventional radiologists [251, 159], one aspect unites them all: the viewing of medical images on conventional monitors [287, 298, 91].

In many of the aforementioned interventional scenarios, real time images are acquired to guide the procedure. However, these images can only be viewed on designated stationary monitors. The ability to position these displays is limited due to sterility, flexibility as they are bound to mounts, and the spatial constraints of the room such as the operating team and equipment. Consequently, images designated for procedural guidance cannot be displayed "in-line" with the operative field [33, 287, 298]. This indirect visualization with images visually off-axis from the intervention site has been shown to create a disconnect between the visuo-motor transformation hindering hand-eye coordination [286]. Situations that allow for the viewing of one's hands and the guiding image simultaneously with an "in-line" view helps to solve this problem [93, 60, 287, 39]. To alleviate this problem, previous approaches placed miniature LCD displays close to the intervention site [33, 287] or displayed images via Google Glass [39, 301]. Unfortunately, in all these cases, the small size and poor resolution of these displays limits the conveyable information impeding standalone use and, hence, clinical relevance [298].

OST-HMDs offer high resolution, binocular displays directly in the field of vision of the user without obstructing the rest of the visual scene [124, 110]. Coupled with medical imaging, this technology may provide virtual monitors that can be positioned

close to the intervention site and are large enough to convey all required information. This technology has the potential to overcome aforementioned drawbacks. Our hypothesis is that the use of virtual displays based on OST-HMDs that enable "in-line" image guidance will allow clinicians to perform procedures with higher efficiency and with improved ergonomics over conventional monitors.

4.2 Contributions

The contributions of this chapter are:

1. We develop a surgical AR application, "virtual monitor" in image-guided surgery, using OST-HMD and real-time medical image streaming. The "virtual monitor" supports various modes of visualization in the space of the operating room, catering to different clinical needs. The details are in Sect. 4.3. Dr. Bernhard Fuerst initiated the idea, and Dr. Mathias Unberath further refined the concept. Kevin Yu assisted in the development of the application.
2. We evaluate the application of the "virtual monitor" in percutaneous spine procedures with phantoms. The procedures include Vertebroplasty, Kyphoplasty, and Disc Decompression. The studies were mainly conducted by Dr. Mathias Unberath and Dr. Gerard Deib, detailed in Sect. 4.5.
3. We develop a set of criteria for evaluating OST-HMDs for the "virtual monitor" setup. The criteria include contrast perception, text readability, task load,

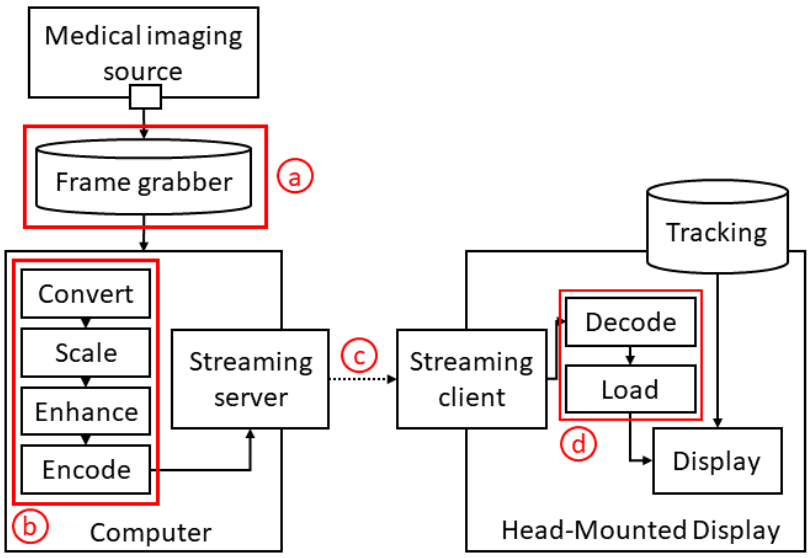


Figure 4.1: The system components and key functionality of a "virtual monitor". Colored annotations highlight functionality modules that are potential performance bottlenecks.

frame rate and system lag. We use the criteria to compare three OST-HMDs, HoloLens 1st gen, Moverio BT-200 and ODG R-7, in Sect. 4.6. Dr. Bernhard Fuerst proposed the concept and contribution. Alexander Barthel contributed in the evaluation, data analysis, and paper writing.

4.3 The Framework of "Virtual Monitor"

The concept of a virtual monitor for IGS can be realized via real-time streaming of the intra-procedurally acquired medical images or image sequences, i. e. video, to an OST-HMD. The OST-HMD then visualizes the images, blending them with the reality perceived by the user. The components of the virtual monitor are demonstrated in Fig. 4.1. In the case presented here, we assume that the OST-HMD is equipped

with a tracking module. Then, medical images can be displayed in different modes allowing for different mixed reality experiences. A more detailed description of the visualization modes is given in Sect. 4.4.

4.3.1 Components

4.3.1.1 Medical Imaging Source

Medical imaging sources provide input to the proposed real-time streaming pipeline. Potential medical imaging sources include 2D imaging modalities such as diagnostic X-ray fluoroscopy systems, interventional C-arm cone-beam scanners, ultrasound systems, and 3D image sources including computed tomography, cone-beam computed tomography, and magnetic resonance imaging.

Traditionally, medical images are transferred within a vendor-specific framework inside the operating room. We use a video output port provided by the manufacturer to tap the medical imaging data after internal pre-processing that is simultaneously supplied to the traditional radiology monitors.

4.3.1.2 Frame Grabber

The frame grabber is hardware that is connected to a video output port of any imaging source (in this case a medical imaging modality) and has access to the imaging data. In setups where the medical imaging source provides an interface for direct

access to the data, the frame grabber is not a necessary component. However, use of a frame grabber has the additional benefit that it effectively decouples medical image generation and internal pre-preprocessing and the proposed streaming pipeline into two separate closed loops, such that the traditional imaging pipeline in the operating room remains unaffected.

4.3.1.3 Image Processing Framework

The image processing framework is responsible for converting, scaling, enhancing, and encoding the image at runtime. Memory-inefficient pixel formats can be converted to more efficient pixel formats that allow for faster processing and transfer, e. g., a conversion from RGBA32 to YUV2, or to gray-scale. Scaling refers to the manipulation of the pixel size of the image and constitutes a trade-off between processing load and image quality. Enhancing is an optional step in the image processing pipeline. Well-known representatives of image processing filters are, e. g., contrast enhancement or denoising [237], that can be employed to further improve the perception and readability of the visualized medical images. Encoding describes the process of compressing images or fragmenting the data into smaller packets to enable efficient transfer or storage. Common encoders include, among others, Motion-JPEG and H264. Motion-JPEG is used in our setup.

4.3.1.4 Data Transfer Network

Data packets are transferred from the image processing framework to the head-mounted display via a data transfer network. Depending on the specifications of the particular image processing framework and HMD device, the data transfer may happen locally, via cable, or via wireless router. For the setup described here we assume use of an untethered device. The image processing framework is realized on a stationary computer and, consequently, a wireless router (NETGEAR Nighthawk R6700) is used for communication and data transfer. TCP/IP acts as the communication protocol.

4.3.1.5 OST-HMD for Visualization

The HMD receives the data packets from the data transfer network, decodes the data packets into images, and loads the decoded images into the rendering engine. Finally, it visualizes the sequence of images in an AR environment with the help of tracking module. Within our experiments, the Microsoft HoloLens 1st gen is used as the OST-HMD.

4.3.2 Tracking and Localization

Tracking and localization is the enabling mechanism for different AR visualization modes. The virtual monitor effect requires the OST-HMD to maintain knowledge

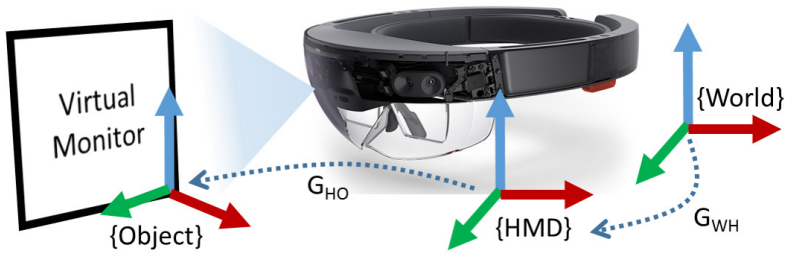


Figure 4.2: Relevant transformations for a “virtual monitor"

about its pose in the OR, involving both hardware sensors and software algorithms. The transformations between the world coordinate systems, OST-HMD and visualized object (here the virtual monitor) are demonstrated in Fig. 4.2. Particularly, G_{WH} is the transformation from the world to the HMD coordinate system and is computed from the tracking module. G_{HO} describes the mapping from the HMD to the virtual object coordinate that is rendered on the OST-HMD; it is controlled by the rendering algorithm. With Microsoft HoloLens, G_{WH} is computed from SLAM algorithms [206] and is available in real-time.

4.4 Visualization of “Virtual Monitor”

Aided by tracking and localization methods, the user is free to choose between three kinds of visualization modes that are presented in this section. This flexibility in AR experience is one of the major advancement of the proposed system compared to earlier systems [39, 94].

4.4.1 Head-Anchored Visualization

In the head-anchored visualization, the rendered object is placed at a fixed pose relative to the user. This means that G_{HO} remains constant. Consequently, medical images are visualized in a **heads-up display** manner. Researchers have exploited the benefits of head-anchored visualization in [269, 39, 94]. Head-anchored visualization is powerful as it makes full use of the HMD in terms of visualization of the content. However, for cases where the surgeon does not want the medical images occluding potentially crucial areas of the operating field, head-anchored visualization is a distraction.

4.4.2 World-Anchored Visualization

The world-anchored visualization is closest to current clinical practice. It creates a virtual monitor effect, where the user is able to see a medical imaging display as if it was presented on a traditional monitor. The 6 degree-of-freedom pose of the virtual object is invariant in the world coordinate system, i. e,

$$G_{WO}(t) = G_{HO}(t) \cdot G_{WH}(t) = \text{const.}$$
$$G_{HO}(t) = G_{WO}(t) \cdot G_{WH}(t)^{-1},$$

where t denotes the current time.

The rendering framework needs to incorporate real-time tracking results, and adjusts the pose of the virtual object accordingly. World-anchored visualization within the medical context has been studied in [37, 273].

World-anchored visualization is intuitive as it resembles the traditional monitor, and gives more control to the user in terms of the display configuration, e. g., the location, orientation, and brightness.

4.4.3 Body-Anchored Visualization

Body-anchored display is a concept that blends both head-anchored and world-anchored display. When the extent of the user's motion is large, the rendered virtual object follows the user's motion similar to head-anchored visualization. While the virtual object remains in the field-of-view of the user at all times, it is not necessarily always at the same pose as it would be in a head-anchored display. On the other hand, when the motion is small, which often happens when the user is slightly adjusting the viewing perspective to better perceive the virtual object, the virtual object remains fixed in the world space as a world-anchored display.

4.5 Virtual Monitor for Percutaneous Spine Procedures

During spine procedures, adequate image guidance is necessary to reliably visualize anatomic landmarks and successfully deliver medical devices. As introduced in Sect. 4.1, radiography monitors displaying the fluoroscopic images used for guidance

Figure 4.3: The operator using the OST-HMD (HoloLens 1st gen) with virtual monitor visualization in the angiography suite.

purposes are typically not aligned with the procedural axis, rendering an indirect visualization shown to hinder hand–eye coordination. In this subsection, we integrated a virtual monitor system based on OST-HMD with percutaneous spine procedures (Fig. 4.3).

4.5.1 Clinical Background

Three routinely performed percutaneous procedures, vertebroplasty, kyphoplasty, and discectomy, were selected in order to demonstrate the feasibility of this novel visualization approach. Percutaneous vertebroplasty (PV) consists of injection of polymethylmethacrylate (PMMA) into fractured vertebral bodies; it is frequently used for the treatment of osteoporotic or metastatic lesions [43, 241, 61]. Kyphoplasty differs from PV by first creating an intravertebral cavity to attempt more controlled

PMMA delivery [276, 300]. Various percutaneous disc decompression (PDD) techniques have been proposed as alternatives to open surgical disc decompression, based on the premise that a reduction in central nucleus pulposus volume decreases the intradiscal pressure and results in retraction of the herniated fragment [187, 139, 145].

The procedural steps for KV, Kyphoplasty and PDD are listed in the following subsections.

4.5.1.1 Procedural Steps for KV

1. Posteroanterior, lateral, and oblique working projections were stored.
2. A 13 g needle was advanced into the anterior third of the vertebral body through the left pedicle using oblique and lateral projections to monitor the progression of the needle and ensure adequate placement without encroaching the medial or inferior aspect of the pedicle.
3. PMMA was prepared (Autoplex Cement Delivery System and half-dose Verteplex cement; Stryker Corporation, Kalamazoo, Michigan, USA).
4. The needle stylet was removed and PMMA injected under posteroanterior and lateral visualization.
5. After PMMA administration, the stylet was replaced under lateral visualization and the needle withdrawn.

4.5.1.2 Procedural Steps for Kyphoplasty

1. Same as for PV (see above).
2. Same as for PV (see above).
3. Once the appropriate needle tip position was reached, the stylet was removed and the kyphoplasty balloon advanced into the anterior third of the vertebral body.
4. The balloon was inflated (with iodinated contrast) under lateral plane visualization. After complete inflation, the balloon was deflated and withdrawn.
5. PMMA was prepared.
6. PMMA was injected under posteroanterior and lateral visualization.
7. After PMMA administration, the stylet was replaced under lateral visualization and the needle withdrawn.

4.5.1.3 Procedural Steps for PDD

1. Posteroanterior, lateral, and oblique working projections were stored. The craniocaudal and lateral angulations on the A plane were adapted to optimize disc space visualization.
2. A 13 g needle was advanced into the center of the intervertebral disc.
3. The stylet was removed and the Dekompressor device (Stryker Corporation) advanced into the disc. Once the tip of the device was adequately located, the

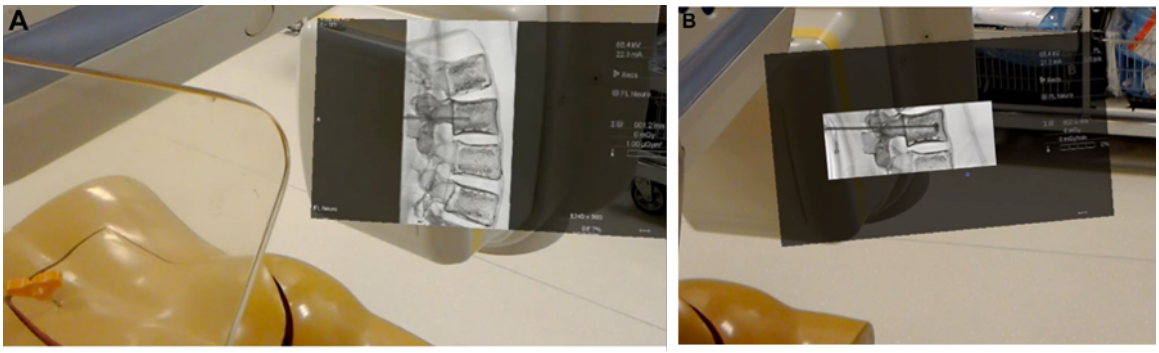


Figure 4.4: Screen capture from HoloLens. (A) Both the virtual monitor and the interventional field are well demonstrated in a single field of view, demonstrating advancement of the vertebroplasty needle. (B) Having the virtual monitor in the same field of view as the operative field allows for close observation of cement placement without the operator turning his/her head.

Dekompressor was turned on to simulate the removal of nucleus pulposus.

4.5.2 Experiment

4.5.2.1 Experiment Setup

The study was performed in a biplane angiography suite (Artis Zee, Siemens Healthcare GmbH, Forchheim, Germany). The operator was a neurointerventional fellow with one year's prior experience of vertebroplasty and kyphoplasty (approximately 25 procedures as the primary operator). The OST-HMD (Microsoft HoloLens 1st gen) created a virtual monitor by superimposing virtual posteroanterior and lateral projections onto the interventionalist's field of view (Fig. 4.4).

In order to facilitate a comparison between traditional and virtual monitor visualization, the key portions of each procedure were repeated four times, once using

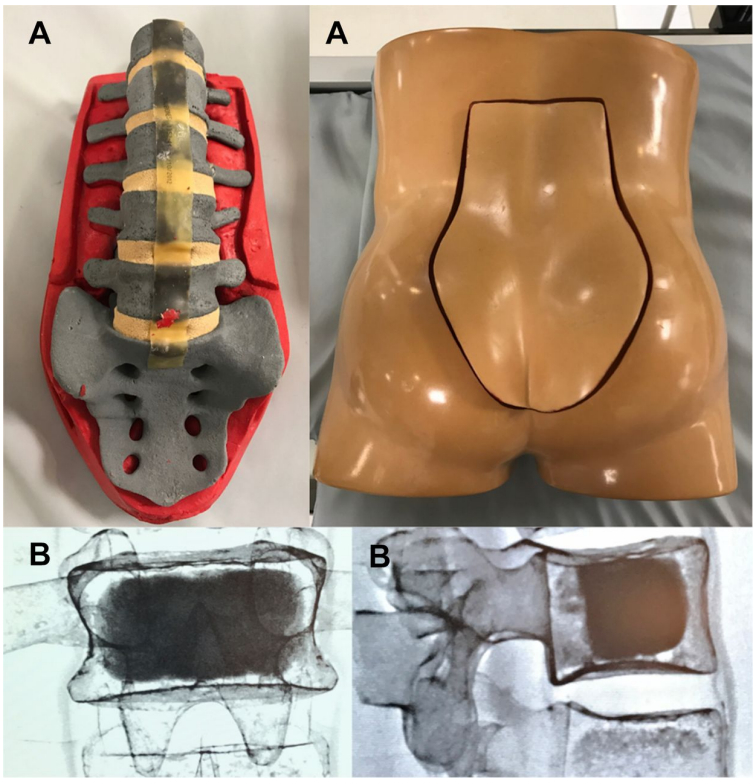


Figure 4.5: (A) Spine model utilized to perform the study. (B) Postprocedural anteroposterior and lateral views demonstrate adequate cement filling of the vertebral body

the standard angiographic display and once with each HMD visualization mode. The order of visualization was randomized. PMMA preparation and injection was performed only once each for PV and kyphoplasty, using a combination of head-anchored and world-anchored modes. The angiography suite monitors were blinded when the procedures were conducted under OST-HMD guidance. The procedures were performed by a single operator on a lumbar spine phantom using commercially available PV, kyphoplasty, and PDD kits (Fig. 4.5).

4.5.2.2 Experiment Evaluation

The following time points were recorded:

1. Time from the start of the procedure to establishing and recording all projections, including the oblique working projection (step 1).
2. Time to place and stabilize the needle in the appropriate access axis for 'end on' transpedicular advancement on the working projection (steps 2 and 3).
3. Time to advance the tip of the needle into the anterior third of the vertebral body with the tip crossing the midline (steps 4 and 5) or into the center of the disc.
4. Time to inflate the kyphoplasty balloon or complete utilization of the Dekompressor device.

When using the virtual monitor, the procedure was filmed through the operator's point of view using each of the three visualization modes: **head-anchored**, **world-anchored** and **body-anchored**. Video recordings were reviewed to assess whether key anatomic landmarks could be consistently and reliably visualized. Procedural dosimetry and duration were recorded. The operator completed a descriptive, qualitative questionnaire following the procedure, detailing the benefits and limitations of each visualization mode. The questionnaire included questions regarding which virtual monitor visualization mode(s) was/were optimal for each portion of the procedure, their ease of use, and comfort of the operator.

4.5.3 Results

All simulated procedures were performed successfully. PV and kyphoplasty were technically adequate, with PMMA filling 60–70% of the L2 and L3 vertebrae, endplate to endplate deposition, an equal amount of cement on either side of the midline, but no unwanted extension towards the posterior quarter of the vertebral body, posterior elements, or extra-vertebral structures. The Dekompressor device was successfully placed into the mid portion of the L3–4 intervertebral disc.

4.5.3.1 Visualization Modes

Head-anchored, world-anchored, and body-anchored visualization modes provided equally effective image guidance. Each mode offered specific advantages depending on the portion of the procedure.

Head-anchored mode was especially useful when utilizing different visual spaces—for example, at the beginning of the procedures, when the operator must look at the patient (or phantom), the angiography C-arm, and the angiography table control panel.

World-anchored mode proved particularly helpful to work in a single visual space, notably when advancing a needle or device under fluoroscopic guidance.

Body-anchored mode combines the convenient features of both the world-anchored and head-anchored modes. However, the constant repositioning of the virtual monitor within the operator's field of view requires a period of acclimation to the mixed reality environment.

4.5.3.2 Visual Landmarks

Key anatomic landmarks, devices, and material components were reliably visualized using both the conventional and virtual environment (in all three visualization modes). During the initial set-up and planning, the operator was able to visualize the superior and inferior endplates of the targeted levels and obtain a working view that clearly outlined the margins of the pedicle and the articulating processes of the facet joints. The needle tip (including the bevel orientation) and the pedicle margins were consistently visualized in both selected working projections; in addition, the lateral view delineated the posterior elements (including the pedicle trajectory), while the posteroanterior view identified the needle tip position in regard to the superior and inferior endplates. During PMMA injection, the cement was adequately visualized both within the needle and the vertebral body. The distribution of PMMA was adequately controlled and extravertebral deposition avoided.

4.5.3.3 Procedural Duration and Dosimetry

Table 4.1: Dosimetry for vertebroplasty procedures

	Monitor	Head	Body	World
Fluoroscopy Time	0.6 min	0.7 min	0.8 min	0.7 min
Dose Area Product (AP)	0.78	1.62	1.28	1.21
Dose Area Product (Lateral)	0.5	0.67	0.94	0.95

The key procedural steps (as listed above) were repeated four times for each

Table 4.2: Procedural times for vertebroplasty procedures

	Time	Monitor	Head	Body	World
Targeting pedicle level	Time 1	01:22.5	01:57.1	01:30.7	02:19.5
Localize needle on skin	Time 2	01:02.8	00:56.1	01:06.4	01:41.1
Needle tip in vertebra	Time 3	02:15.7	01:56.6	02:22.1	01:39.7
	Total	04:41.0	04:49.8	04:59.2	05:40.3

Table 4.3: Dosimetry for kyphoplasty procedures

	Monitor	Head	Body	World
Fluoroscopy Time	1.6 min	1.9 min	1.3 min	1.2 min
Dose Area Product (AP)	1.07	1.49	1.55	1.5
Dose Area Product (Lateral)	3.47	2.98	2.42	2.13

procedure, first using the standard angiographic display and then three times with virtual monitor, using each of the visualization modes. The results are summarized in the following tables. Total procedural times, key intra-procedural times, beam time, and dose area product measurements were similar when comparing virtual monitor to the traditional monitor for vertebroplasty (Tab. 4.1 and Tab. 4.2), kyphoplasty (Tab. 4.3 and Tab. 4.4), and disc decompression (Tab. 4.5 and Tab. 4.6).

4.5.3.4 Operator Preferences and Observations

While all procedural steps and salient structures were adequately visualized using all three visualization modes with virtual monitor, there was a significant learning

Table 4.4: Procedural times for kyphoplasty procedures

	Time	Monitor	Head	Body	World
Targeting pedicle level	Time 1	01:36.6	02:10.0	01:36.0	01:17.5
Localize needle on skin	Time 2	01:31.1	01:21.6	01:04.9	01:10.3
Needle tip in vertebra	Time 3	02:07.4	02:10.3	01:57.8	01:33.6
Balloon inflated	Time 4	03:28.9	02:42.8	01:18.7	01:17.2
	Total	08:44.1	08:24.7	05:57.4	05:18.6

Table 4.5: Dosimetry for disc decompression procedures

	Monitor	Head	Body	World
Fluoroscopy Time	0.6 min	0.5 min	0.5 min	0.7 min
Dose Area Product (AP)	2.66	1.19	1.17	1.48
Dose Area Product (Lateral)	0.5	0.87	1.02	1.43

curve in becoming familiar with OST-HMD's controls and getting used to the presence of the virtual monitor. Initially, the world-anchored mode was preferred, being the closest to a standard monitor and therefore more intuitive. However, the benefits of the body-anchored mode became apparent as the operator grew familiar with the AR environment and functionalities. After an adjustment period, the operator found the OST-HMD unobtrusive and was able to wear it throughout the procedure without discomfort and without significant field of view impairments during any procedural step using any of the visualization modes. The motion of the display in the body-anchored mode was occasionally found to be disorienting, particularly during placement and advancement of the needle.

Table 4.6: Procedural times for disc decompression procedures

	Time	Monitor	Head	Body	World
Targeting disc level	Time 1	02:28.3	02:29.3	01:51.4	02:08.6
Localize needle on skin	Time 2	01:08.4	01:22.6	01:03.3	01:18.9
Needle tip in nucleus	Time 3	03:18.8	01:07.3	01:54.6	02:11.6
Complete device exchange	Time 4	01:14.2	00:35.9	00:37.8	00:40.0
	Total	08:09.7	05:35.1	05:24.8	06:19.2

4.5.4 Discussion

We demonstrated the feasibility of performing percutaneous spine interventions using a virtual monitor (without the use of traditional monitors). In this single user preliminary study, virtual monitor visualization was not found to be inferior to traditional monitors in terms of procedural duration, dosimetry, or ability to visualize key anatomic structures, devices, or material components. There was no perceived impairment in the operator's ability to visualize real world structures while wearing the OST-HMD and using the virtual monitor display, as demonstrated by the completion of all procedural steps without removing the OST-HMD, turning off the virtual monitor, or resorting to a traditional monitor.

A virtual monitor for a percutaneous procedure performed in the angiography suite carries several potential advantages. First, the key imaging plane(s) and the procedural site are constantly present in the operator's field of view. This may be in a single fixed location (as in world-anchored visualization) or following the operator's

field of view in other modes. Operators do not need to modify their position, significantly rotate their head, or move their field of view away from the procedural site in order to obtain image guidance information. In addition, the operator can independently adjust display characteristics (e.g., location, size, contrast, luminosity) using hand gestures or voice control, without relying on other team members to adjust the display's position or characteristics.

OST-HMDs may also prove advantageous in percutaneous procedures taking place in other environments—for example, intraoperative angiography performed in operating rooms using mobile radiographic equipment and displays, which are often cumbersome and placed in ergonomically challenging positions for the operator due to space limitations. The potential role of OST-HMDs might even be expanded further if the possibility of conducting procedures outside angiography or operating rooms is considered. Having a compact, lightweight, wearable, and easily transportable display opens the door for image-guided percutaneous procedures to be performed in an outpatient clinic setting or at the bedside, for example, taking advantage of constantly smaller, more mobile image acquisition equipment, including in-office fluoroscopy, ultrasound, or even CT/MRI scanners.

4.5.5 Summary

In this section, we implemented a virtual monitor system for percutaneous spine procedures. The preliminary study demonstrated the feasibility of using a virtual

monitor as an alternative to conventional physical displays for image-guided procedures. This novel visualization approach may represent a valuable adjunct tool for minimally invasive percutaneous procedures in general, notably when performed in spatially limited environments.

4.6 Criteria for Choosing OST-HMD for Virtual Monitor

In the previous section, we developed a virtual monitor system with Microsoft HoloLens. However, since there are many OST-HMD products in the market, it is unknown to the developers or clinicians which OST-HMD should be used for specific procedures. In this section, we present a systematic approach to identify the criteria for evaluation of OST-HMDs for AR guidance. We limit the visualization method to **object-anchored** visualization, which is a specific type of world-anchored visualization, where the virtual monitor is registered to a tracked object. With object-anchored display, medical information can be displayed close to the desired object, for example, the surgical site.

4.6.1 Proposed Evaluation Criteria for OST-HMDs

Criteria for evaluating OST-HMD devices are proposed in this subsection: text readability, contrast perception, task load, frame rate, and system lag. For generality, the impact of procedure-dependent issues such as OR lights is not considered.

4.6.1.1 Text Readability

The patient demographics, diagnostic information, and vitals are usually displayed as plain text. Therefore, it is necessary to evaluate how well the user is able to perceive text displayed on the OST-HMD. Text readability is user-dependent and is affected by the screen resolution, screen refresh rate, and blur introduced by the optics design.

4.6.1.2 Contrast Perception

It is important for the surgeon to be able to distinguish even slight differences in contrast in medical images in order to facilitate decision making during the intervention. Therefore, contrast perception is proposed as one of the metrics for evaluation. Similar to text readability, contrast perception is affected mainly by the optics capability and is user-dependent [115].

4.6.1.3 Task Load

OST-HMDs may aid users during intervention but also impose extra task load. The task load for OST-HMD visualization is affected by the ergonomics of the OST-

HMD, the duration of the task, and the eye fatigue caused by the display [104], etc. NASA-TLX [97] is chosen for the assessment of task load.

4.6.1.4 Frame Rate

Frame rate is critical to comfortable perception and smooth use of OST-HMDs [231]. Augmentations rendered with low frame rate cause an unpleasant experience for users. Frame rate is a comprehensive measure of the hardware capability of the OST-HMDs, and can be measured by profiling the application.

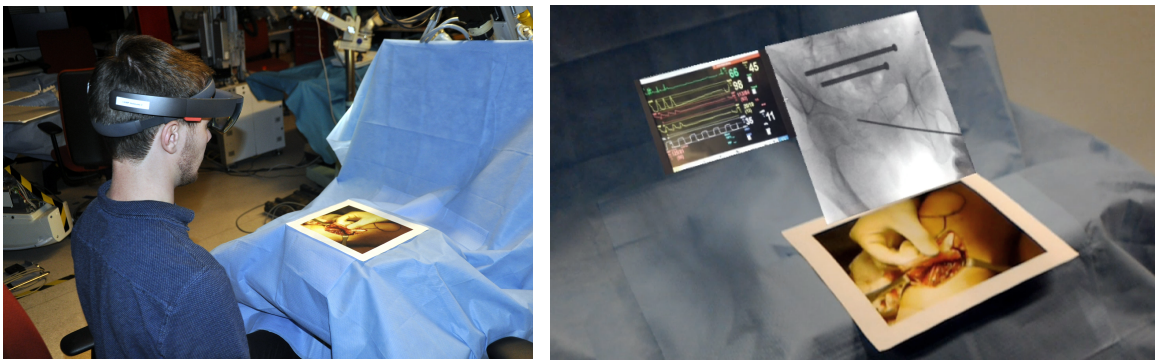
4.6.1.5 System Latency

The system lag is the combination of the time spent on tracking, rendering, and visualization. High system lag causes unpleasant experience for the user as well, especially in terms of incorrect registration between virtuality and reality. The measurement of system lag usually requires a more capable testing platform.

4.6.2 Experiment

4.6.2.1 Experiment Setup

A combined comparative study, involving a multi-user study for subjective criteria, and an offline experiment of system capability, is set up in order to evaluate the performance of three OST-HMDs for object-anchored virtual monitor during inter-



(a) Third-person view **(b)** First-person view

Figure 4.6: (a) Participant stands beside the simulated surgery site. An image is used as a tracking target. (b) A virtual monitor is anchored to the tracking target.

ventions.

The three off-the-shelf OST-HMDs are: Epson Moverio BT-200, ODG R-7, and Microsoft HoloLens. Each device uses a different display technology (projector-based, LCD projector-based, and holographic waveguide). A summary of the hardware comparison is listed in Tab. 1.1 and Tab. 1.2 in Sect. 1.1.2.

An image simulating an orthopeadic surgery scene is attached to a blue drape, serving as a tracked target (Fig. 4.6b). The user wearing the OST-HMD is standing at a marked position, looking down onto the simulated surgery site. The image displayed on the virtual monitor is controlled by the researcher. It has a physical size of 20×20 cm. The system setup is illustrated in Fig. 4.6.

Sample images for the evaluation of **text readability** and **contrast perception** are demonstrated in Fig. 4.7 and Fig. 4.8 respectively. A short sentence with varied font size (denoted f) is placed on the 1024×1024 black background. The font size is displayed in the top left corner. Each image for the evaluation of contrast perception

Figure 4.7: Three sample images for evaluating the text readability.

Figure 4.8: Three sample images for evaluating the contrast perception.

contains four shapes with different directions. The size of the shape is 200×200 pixels, and the size of the background 1024×1024. The contrast value (denoted c) of the current image is displayed in the top left corner as well. The actual grayscale value of the shape is $1 - c$.

4.6.2.2 Multi-User Evaluation

Each participant performs the experiment with all three of the OST-HMDs in a random order, minimizing the learning bias. While subjectivity in the test criteria

exists, e.g., eyesight, it contributes equally for each device yielding minimal bias toward a particular device. Several series of images are presented sequentially on the virtual monitor for evaluation of the subjective criteria proposed in Sect. 4.6.1. The procedure of the experiment for each participant is:

1. The participant fills out a consent form and pre-experiment survey.
2. The subject is shown a series of 10 short sentences on a transparent background. The subject is asked to read the sentences out loud to make sure the system is working well and the user is perceiving the test images correctly.
3. Shapes with decreasing contrast value c are displayed (Fig. 4.8) to the participant. The subject has to identify the directions of the shapes. The smallest contrast value c_{min} at which the participant is still able to tell the directions of the shapes is recorded.
4. The subject is shown a series of short sentences with decreasing font size f (Fig. 4.7). The smallest font size f_{min} is recorded for which the subject is still able to correctly read the text.
5. The subject fills out the NASA-TLX form for the OST-HMD being used.
6. Steps 2-5 are repeated for the other two OST-HMDs.

The user study was conducted with 20 participants between the ages of 22 and 46. Participants were recruited from non-medical (13) and medical (7) students.

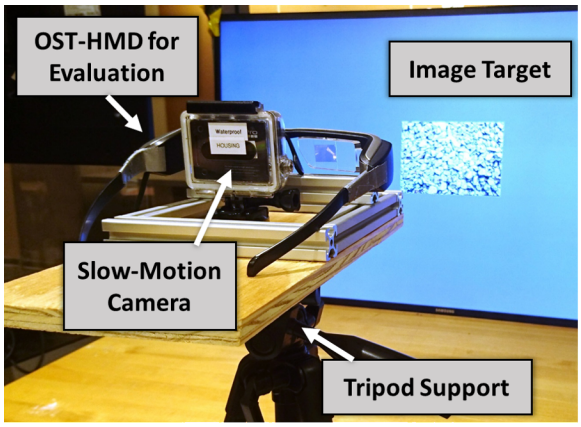


Figure 4.9: Experimental setup for offline evaluation of system lag. The slow-motion camera captures the motion of the image target and the display on the OST-HMD.

4.6.2.3 Offline Evaluation

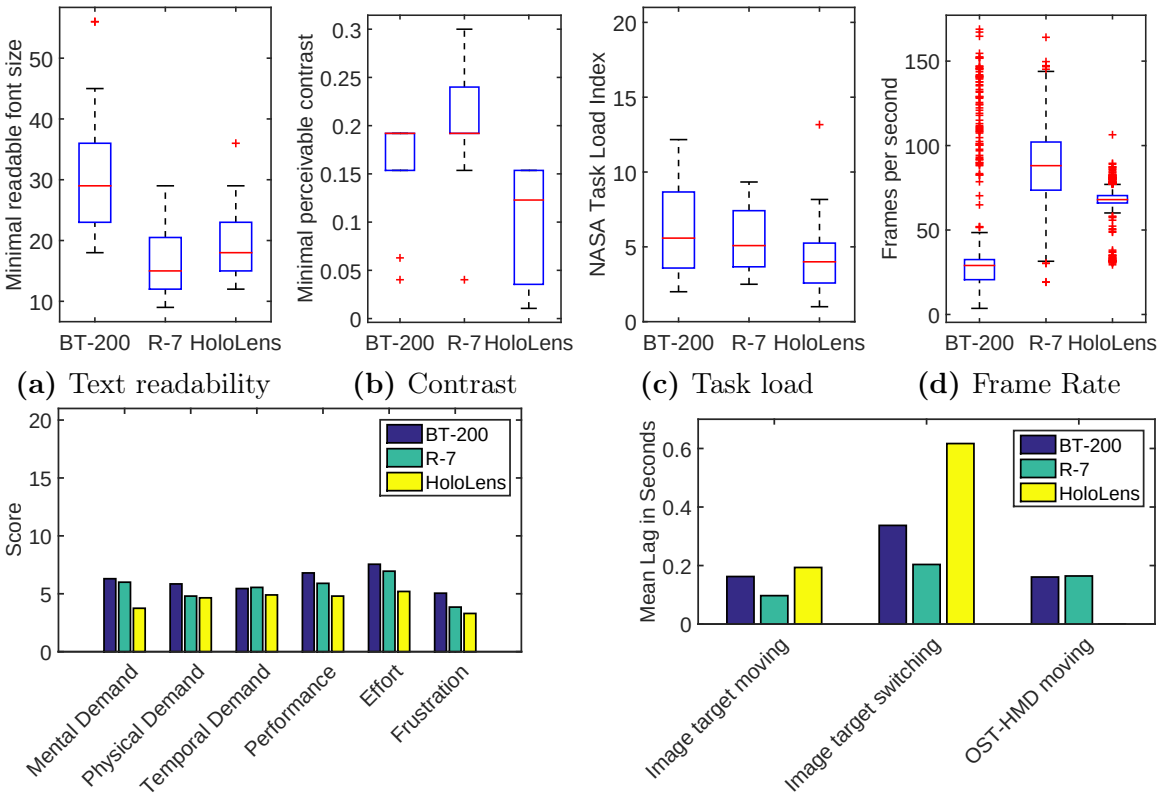
An offline experiment is set up to evaluate the system lag of the OST-HMDs. The OST-HMD under evaluation is mounted on a tripod. A large screen showing the image target is located in front of it. The slow-motion camera (GoPro Hero 4) is placed behind the OST-HMD, capturing both the motion of the image target and the response on the OST-HMD. The system setup is shown in Fig. 4.9. The system lag for each OST-HMD is measured in three experimental situations: the image target is moving on a defined path, the image target is suddenly switching position, and just the OST-HMD is moving. Each experiment is repeated 16 times. The system lag is measured by calculating the temporal difference between the change of the environment and the response of the system, via manual annotation.

4.6.3 Results and Discussion

Results of the user study are shown in Fig. 4.10. A statistical analysis was performed to study the differences of the devices in the experimental setting described. Significance is achieved for p-values lower than 0.05. Normal distribution of the data is not assumed, therefore, the Friedman test is performed. The test shows significant differences in the smallest readable font size $\{f_{min}\}$ ($\chi^2(2) = 27.26, p = 0.01$), the minimal distinguishable contrast value $\{c_{min}\}$ ($\chi^2(2) = 27.24, p < 0.01$), and NASA-TLX ($\chi^2(2) = 16.95, p < 0.01$) with respect to the OST-HMD being used.

4.6.3.1 Text Readability

The post-hoc tests are performed using the Wilcoxon signed-rank test [289]. A Bonferroni correction for multiple comparisons is applied. Comparing each of the devices to the other ones for each of the three results yields nine hypotheses. The statistical evaluation tests $n = 9$ hypotheses with a desired level of significance $p = 0.05$. The p-values to test against are therefore adjusted to $p/n = 0.05/9 = 0.0056$. The resulting z-scores are then compared to the critical z-value (2.753) for the given level of significance. To reject the null hypothesis, which says that there is no difference between the two devices, the z-score has to be greater than the absolute of the critical z-value for the given level of significance. The post-hoc tests show a significant improvement of text readability from BT-200 to R-7 ($Z = 2.930, p = 0.0056$), and HoloLens yields better results compared to BT-200 ($Z = 3.510, p = 0.0056$). How-



(a) Text readability **(b)** Contrast **(c)** Task load **(d)** Frame Rate

(e) Results of the NASA-TLX

(f) Lag measured in the offline experiment

Figure 4.10: Experiment results of the proposed criteria for three OST-HMDs. Lower values indicate better performance.

ever, there is no significant difference between HoloLens and R-7. One of the possible reasons for HoloLens and R-7 to perform better than BT-200 is the higher screen resolution and screen refresh rate.

4.6.3.2 Contrast Perception

All combinations of BT-200, R-7, and HoloLens show significant differences in the minimal contrast value that is distinguishable. This value is lower for BT-200 than R-7 ($Z = -2.880, p = 0.0056$). There are also significant improvements from R-7 to

HoloLens ($Z = 3.470, p = 0.0056$) and from BT-200 to HoloLens ($Z = 3.039, p = 0.0056$). HoloLens outperforms R-7 and BT-200 in providing correct perception of low contrast images.

4.6.3.3 Task Load

There is a significant reduction in NASA-TLX from BT-200 to HoloLens ($Z = 3.142, p = 0.0056$) and from R-7 to HoloLens ($Z = 2.991, p = 0.0056$). The difference in NASA-TLX between BT-200 and R-7 is not significant. The detailed results of the NASA-TLX questionnaire for each device is visualized in Figure 4.10e.

Although the HoloLens is heavier than the BT-200 and the R-7, its ergonomic design is more adjustable. With correct adjustment, the weight of HoloLens is not imposed on the user's nose, but distributed around the head. However, since the experiment generally lasts 10 minutes for each OST-HMD, the effect of weight is not sufficiently evaluated for time-consuming tasks by our experiment. A few participants held the BT-200 with their hands, which indicates that the ear hook design of the BT-200 may not be sufficient to securely attach the device to the user's head. The relative motion between the OST-HMD and the user's eye may invalidate the display calibration [115].

Eye fatigue is another source of task load for users in HMD-based tasks. Vergence-accommodation conflict is identified as the major cause of visual fatigue [104]. For the projector-based optics on which the R-7 and the BT-200 are constructed, the

accommodation distance is fixed at the distance of the light source, while the vergence distance is about 1 m away from where the tracked object is placed. On the contrary, the HoloLens is a multiscopic display device [133], with reduced conflict between vergence and accommodation [132].

4.6.3.4 Frame Rate

The information about the frame rate of the OST-HMDs is accessed via the Unity3D profiling tool. The BT-200 on average takes 0.0407 s to render a frame (standard deviation 0.0169 s).The frame rate regularly dropped to less than 20 frames per second. This results in noticeable jitter when the user moves his or her head slightly. The average time spent rendering one frame on the R-7 is 0.0124 s (standard deviation 0.0034 s), and the average number of frames per second is 87.6648. The frame rate of the HoloLens device is most stable by observation. Mean render time was 0.0151 s (standard deviation 0.0029 s), which corresponds to 67.7081 frames per second on average. Lower frame rates occurred only occasionally. The real-time performance of each device is illustrated in Fig. 4.10d.

4.6.3.5 System Lag

The average system lag for each device in the three experimental situations is visualized in Fig. 4.10f. In the experiment with the image target moving or switching locations, the system lag of BT-200 (0.163 s and 0.337 s) and R-7 (0.097 s and

0.204 s) is smaller than HoloLens (0.193 s and 0.617 s). However, in the experiment that investigates lag when the OST-HMD is moving rather than the image target, the system lag of HoloLens is within one single frame of the slow-motion camera. Therefore, the lag is not measurable by the experimental setup and is significantly smaller than BT-200 (0.161 s) and R-7 (0.164 s). Both frame rate and system lag are mainly affected by the hardware capability and the tracking modality. Each OST-HMD is equipped with different sensors and algorithms. The joint effort of indoor location and image target tracking implemented by HoloLens results in this unique behavior of the device. For surgical scenarios, which are indoor and do not involve frequent motions of the registration target, HoloLens might be considered a more suitable OST-HMD in terms of system lag.

4.6.4 Summary

With the increasing availability of OST-HMDs in the consumer market and the interest from the clinical community to deploy them [231], it is necessary to propose clinically relevant criteria to evaluate the suitability of different OST-HMDs and conduct a comparison between some commercially available OST-HMD devices. Evaluation criteria for OST-HMDs providing virtual monitor visualization of image-guided surgery are then proposed: **text readability**, **contrast perception**, **task load**, **frame rate**, and **system lag**. Epson Moverio BT-200, ODG R-7, and Microsoft HoloLens were selected to be assessed by our comparative multi-user study.

These devices were chosen as they are representatives of currently available technologies. Twenty participants were recruited for the multi-user study to evaluate the perceptual performance of each OST-HMD, and an offline experiment was conducted to directly evaluate the system lag. Results demonstrate that HoloLens outperforms R-7 and BT-200 in contrast perception, task load, and frame rate. For text readability, there is no significant difference between HoloLens and R-7, and they both outperform BT-200. The integration of localization and optical tracking on HoloLens yields significantly smaller system lag in the situation where the OST-HMD is moving in an indoor environment. Based on our analysis, HoloLens has better performance in the proposed scenario at present. However, the clinical benefit of OST-HMDs during a particular intervention still has to be determined by procedure-specific experiments.

4.7 Conclusion

In this section, we described the "virtual monitor" based on OST-HMD, which is intended for visualization of medical images during image-guided surgeries. The components of a virtual monitor system include the medical imaging source, a frame grabber to receive the images from the current OR setup, an image processing network, a data transfer network and an OST-HMD for visualization. The visualization on OST-HMD can be categorized into head-anchored, world-anchored and body-anchored modes. We specifically implemented a virtual monitor for percutaneous

spine procedures (KV, Kyphoplasty, PDD) and evaluated the feasibility of the novel visualization with phantom studies. With increasing interest of using OST-HMDs for AR display of medical images, the suitability of different OST-HMD products for this specific task needs to be evaluated. Therefore, we proposed three subjective and two objective criteria (text readability, contrast perception, task load, frame rate and system latency), and evaluated three current generation OST-HMDs (HoloLens 1st gen, Epson Moverio BT200, ODG R-7).

4.8 Closing Remarks

Technically speaking, the virtual monitor does not require sophisticated hardware and software, e.g. accurate tracking and registration. A functional virtual monitor can be achieved with the current generation of OST-HMDs. The "simplicity" enables such application to be widely evaluated in terms of the usability and clinical benefits in the real clinical environment.

4.9 Published Work

Material from this chapter appears in the following publications:

1. Gerard Deib, Alex Johnson, Mathias Unberath, Kevin Yu, Sebastian Andress, **Long Qian**, Gregory Osgood, Nassir Navab, Ferdinand Hui, Philippe Gailloud, "Image Guided Percutaneous Spine Procedures using an Optical See-Through Head Mounted Display: Proof of Concept and Rationale," *Journal of Neu-*

rointerventional Surgery (JNIS), Volume 10, Issue 12, pp. 1187-1191. British Medical Journal Publishing Group. 2018.

2. **Long Qian**, Alexander Barthel, Alex Johnson, Greg Osgood, Peter Kazanzides, Nassir Navab, Bernhard Fuerst, "Comparison of Optical See-Through Head-Mounted Displays for Surgical Interventions with Object-Anchored 2D-Display," *International Journal of Computer Assisted Radiology and Surgery (IJCARS)*, Volume 12, Issue 6, pp. 901-910. Springer. 2017.

3. **Long Qian**, Mathias Unberath, Kevin Yu, Bernhard Fuerst, Alex Johnson, Nassir Navab, Greg Osgood, "Towards Virtual Monitors for Image Guided Interventions Real-Time Streaming to Optical See-Through Head-Mounted Displays," *arXiv*, 1710.00808. 2017.

Chapter 5

ARssist: AR for the Bedside Assistant in Robotic Surgery

In this chapter, the development and evaluation of an AR application (*ARssist*) for the bedside assistant in robotic surgery is presented and discussed. *ARssist* takes advantage of an OST-HMD and a da Vinci robotic system, aiming to improve the ergonomics of the assistant, e.g. poor hand-eye coordination caused by the mis-orientation of the monitor display. The experiments demonstrated that *ARssist* is able to significantly improve the performance of inexperienced users, especially under the circumstances of large extent of mis-orientation of the monitor display.

5.1 Introduction

In the previous chapter, we introduced virtual monitor as an alternative to visualize the medical images using OST-HMD. This chapter investigates the use of AR and OST-HMD in robotic surgeries. In the context of robotic surgery, the laparoscopic video provides the main visual guidance, and can be visualized using the virtual monitor, while other critical information, e.g. status of the robotic instruments and robotic-driven endoscope, could be exploited as useful surgical guidance through the transparent screen of the OST-HMD.

In a *da Vinci*® robot-assisted surgery, the main surgeon sits at the console tele-operating the robot, while the patient-side assistant stands or sits at the bedside assisting the operation (see Fig. 5.1). The patient-side assistant, also called bedside assistant, scrubbed surgeon [134], or first assistant (FA) [157], plays an important role in the robotic laparascopic surgery. Before the main surgeon starts tele-operation, the FA is responsible for, or takes an important role in, trocar placement, docking of the robot, and preparing the operative field. During the surgery, the FA exchanges the instrument for the main surgeon, manipulates certain laparoscopic instruments, e.g., gripper and vessel sealer, and extracts specimen [229, 134, 157].

The outcome of a robotic surgery is dependent on the performance of the FA. Through an analysis of 222 urologic cases, researchers have identified that the mean operative time for all robotic procedures showed a consistent trend of reduction with increasing experience of the FA [178]. In another study comparing the performance of

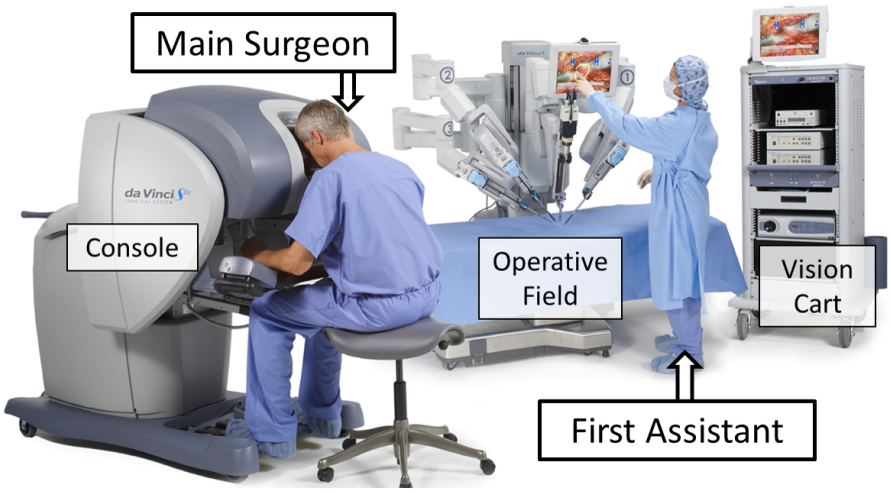


Figure 5.1: Surgery team with a *da Vinci S*® surgical robot; The first assistant is usually at the bedside, assisting the procedure. Image © 2019 Intuitive Surgical, Inc.

well-trained and less-trained FAs among 280 different robotic surgical interventions, the authors concluded that interventions with a well-trained FA are more rapid and secure [229].

The *da Vinci* system restores the hand-eye coordination and depth perception of the surgical field for the main surgeon. But the improvement does not benefit the FA. For example, when the FA needs to install or exchange an instrument for the main surgeon, he/she has to manually and blindly adjust the robotic arm in order for the instrument to appear in the operative field, or have the console surgeon reposition the endoscope to visualize the instrument until it arrives at the desired location. As another example, when the FA is manipulating instruments inside the patient body, he/she has to look at the monitor mounted on the vision cart that is not near the operative field, which leads to an awkward hand-eye coordination.

We propose to use OST-HMD-based AR, to address the aforementioned problems of current laparoscopic robots. We present the system *ARssist*, an application based on the integration of a da Vinci robot and an OST-HMD. *ARssist* provides various AR information to the FA, including: (1) 3D real-time rendering of the endoscope, robotic instruments and hand-held instruments within the patient body, and (2) real-time stereo endoscopy that is configurable for the the FA's preferred hand-eye coordination. We choose two frequent tasks of FA during interventions: instrument insertion (II) and tool manipulation (TM) for evaluation. We have performed 3 iterations of system implementation and evaluation, which will be detailed in this chapter.

5.2 Contributions

The contribution of this chapter is:

1. We develop *ARssist*, an OST-HMD based AR application for the bedside assistant in robotic-assisted surgery and evaluate the user performance during instrument insertion and tool manipulation for both experienced and inexperienced users. *ARssist* significantly improves the hand-eye coordination of the user, especially for less experienced users and in mis-orientation situations. Anton Deguet assisted me by developing software that provides low-latency UDP packet streaming from the da Vinci robot.

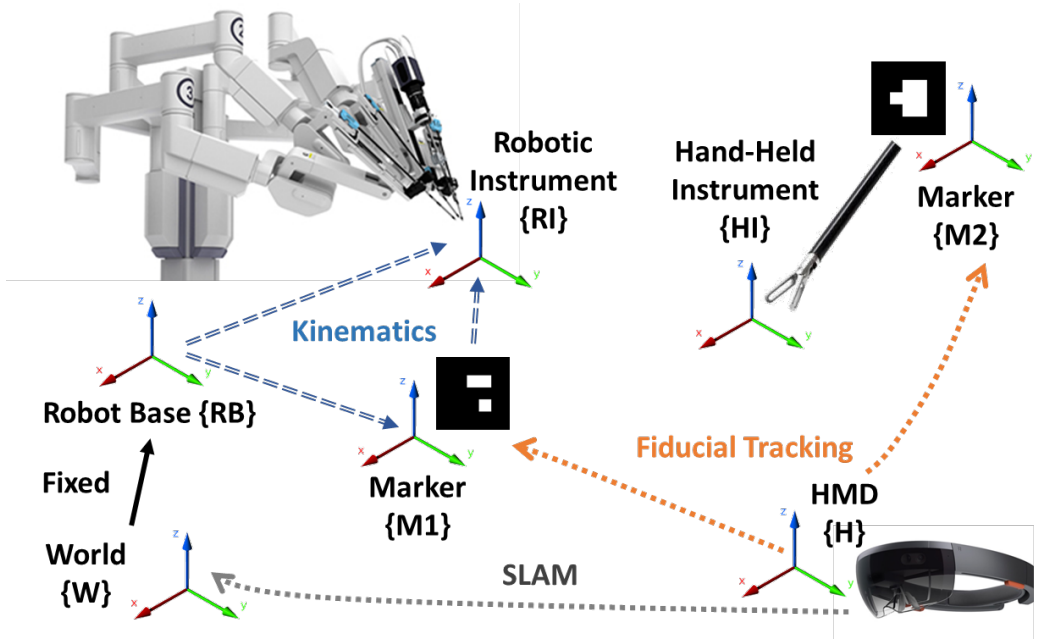


Figure 5.2: Components of *ARssist* and their relative transformations

5.3 Methods

5.3.1 Components and Transformation Map

In order to offer visualization of robotic instruments and hand-held instruments at the correct location and orientation with respect to the viewer, the system must track them in real time. Fig. 5.2 shows some *ARssist* components. We assume that these components are rigid bodies and affix a Cartesian coordinate system to each one.

Different components in *ARssist* are geometrically linked in various ways. In a robotic surgery, once docked the robot remains stationary within the operating room. Therefore, it is safe to assume that the transformation between the world and the

robot base is fixed, i.e., T_W^{RB} is a constant. The robotic instruments are controlled precisely by the robot during the surgery. The transformation between the robot base and robotic instrument, T_{RB}^{RI}, is obtained from the robot model and real-time kinematics data. We attach fiducial markers to certain parts of the robot and to the hand-held instruments to support optical tracking on the OST-HMD. The markers cannot be attached directly to the tool tip because they will not be visible to the HMD during the surgery. As a result, the fiducial markers are 'plugged' into the robot kinematics chain. The poses of the markers are dependent on joints that are closer to the base of the robot. For a robotic instrument, $T_{RB}^{M_1}$ and $T_{M_1}^{RI}$ are both obtained from robot kinematics, and for hand-held instruments, the transformation $T_{M_2}^{HI}$ is fixed. The transformations between the markers and the OST-HMD, $T_{M_1}^{H}$ and $T_{M_2}^{H}$, are computed at runtime via vision-based tracking algorithms. In addition, it is notable that recent OST-HMDs offer inside-out localization (Tab. 1.1 and Tab. 1.2). The OST-HMD can compute T_W^H at runtime through inside-out tracking methods.

Therefore, the transformation between the OST-HMD and a robotic instrument can be computed in two ways:

$$T_H^{RI} = T_{M_1}^{RI} \cdot T_H^{M_1} \tag{5.1}$$

or,

$$T_H^{RI} = T_{RB}^{RI} \cdot T_W^{RB} \cdot T_H^{W} \tag{5.2}$$

The transformation between the OST-HMD and a hand-held instrument is:

$$T_H^{HI} = T_{M_2}^{HI} \cdot T_H^{M_2} \tag{5.3}$$

Eq. 5.1 uses the fiducial tracking, the kinematics data, the model of the robot, and the pivot calibration that determines the pose of the marker relative to a certain joint of the robot. Eq. 5.2 uses the inside-out tracking capabilities of the HMD, the robotic model, kinematics data, and the calibration. Eq. 5.3 uses the fiducial tracking and the pivot calibration. It is notable that there exists redundancy in the tracking of robotic instruments.

5.3.2 Hybrid Tracking Scheme for Robotic Instruments

In *ARssist*, we take advantage of the redundancy and employ a hybrid tracking scheme, derived from [274], to localize the robotic instruments. Our tracking scheme is comprised of three steps. First, the prioritization of each transformation is determined with prior knowledge, so that reliable and accurate transformations are given higher priority. Then, we prioritize different tracking methods, which are constructed by composing transformations with different priorities. These two steps are conducted in an offline stage. Finally, at the online stage, we always use the tracking method of highest priority when it is available. When the highest priority method is not available, e.g., due to line-of-sight loss, we model the discrepancy between the lower and higher priority tracking methods as a static error, and compensate for it when switching from the high-priority tracking method to a low-priority tracking method.

Table 5.1: Transformations and priorities between components of *ARssist*

Transformation	Computation	Priority
World to Robot Base T_W^{RB}	Fixed	High
Robot Base to Robot Inst. T_{RB}^{RI}	Kine. + Model	High
Robot Base to Marker $T_{RB}^{M_1}$	Kine. + Model + Piv.	High
Marker to Robotic Inst. $T_{M_1}^{RI}$	Kine. + Model + Piv.	High
Marker to Hand-held Inst. $T_{M_2}^{HI}$	Piv.	High
Marker to HMD $T_{M_{\{1,2\}}}^{H}$	Fiducial Tracking	Medium
World to HMD T_W^H	SLAM	Low

The transformations that are fixed or derived from kinematics data are given high priorities, because they are most reliable in terms of accuracy and latency. They can be reliably calculated within a few millimeters [135]. Transformations obtained from fiducial tracking are assigned medium priority. The accuracy of fiducial marker tracking will suffer when the relative motion between the OST-HMD and the marker is more significant, as the latency caused by camera exposure and computation is not negligible. Furthermore, the accuracy of camera-based fiducial tracking is affected by the distance from the object to the camera, and specific software algorithm. It could be around several centimeters [1]. At last, we assign low priority to the self-localization of the OST-HMD. Note that we assign these priority levels based on the current generation of OST-HMD hardware and software. We summarize the transformations and the priorities between each component in Tab. 5.1.

5.3.3 Kinematic Streaming

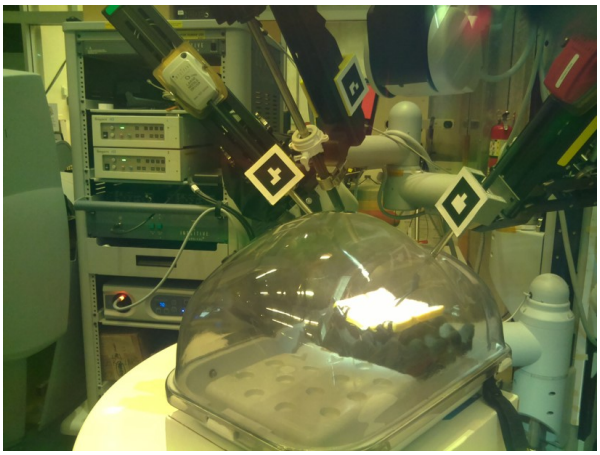
In order to compute the transformation from robot base to robotic instrument, the robotic model and status need to be known to *ARssist*. The robot model, e.g. the DH parameters, could be stored in the AR application beforehand, but the robot status, e.g. the joint values, is constantly being updated. Therefore, the robot status is real-time streamed to *ARssist*. On the OST-HMD, the virtual robot is being configured each time it receives a status message from the robot. The kinematic streaming from a da Vinci robot to a mixed reality application is later open sourced as a contribution to the dVRK community, named dVRK-XR. More information about dVRK-XR is presented in Sect. 5.11.

5.3.4 Visualization of Stereo Endoscopy

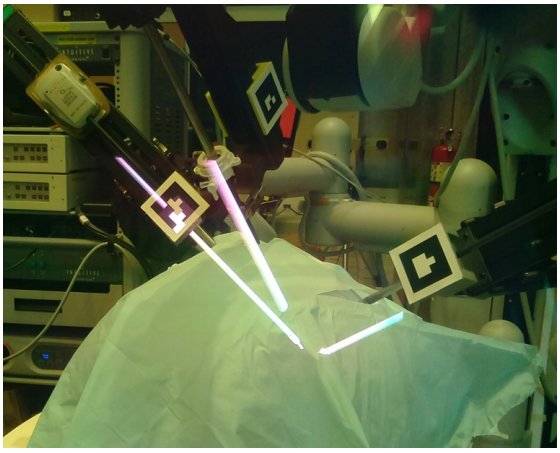
The endoscopy serves as the primary feedback both for the console surgeon and for the first assistant. For da Vinci robot, the endoscopy is binocular. A binocular OST-HMD can present the left and right endoscope channel to the left and right eye, respectively, thereby restoring the depth perception of the endoscopy to some extent.

ARssist offers three visualization options: 1) **head-anchored display**, 2) **world-anchored display**, and 3) **frustum projection**. Both head-anchored display and world-anchored display are adapted from the "virtual monitor" concept from Ch. 4.

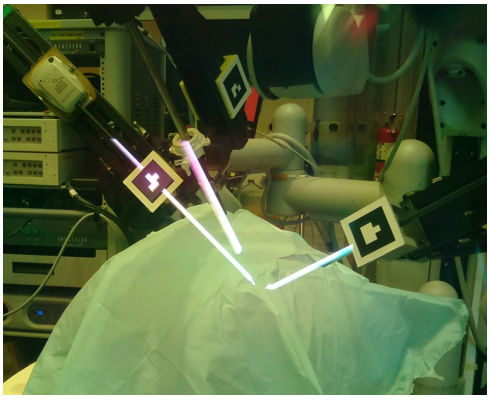
Frustum projection is a novel visualization technique that renders the en-



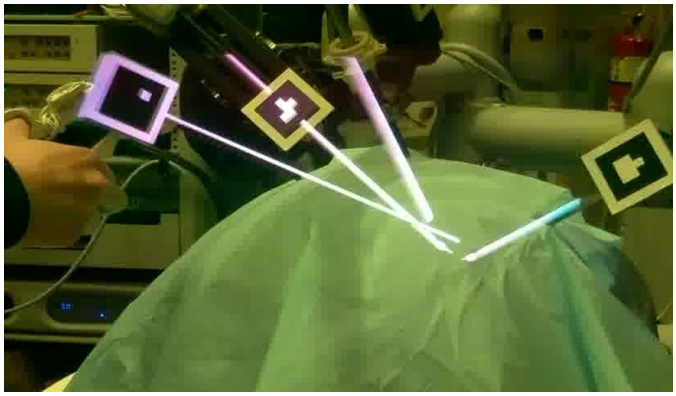
(a) Transparent body phantom



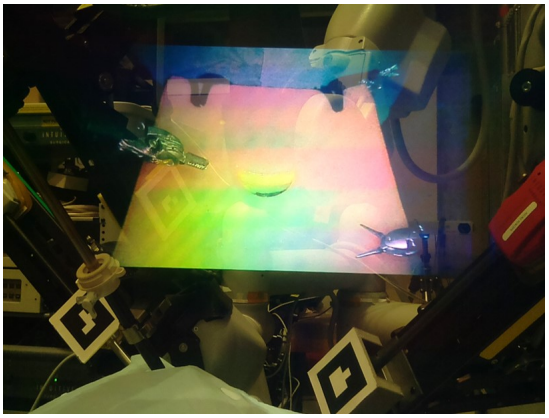
(b) Before display calibration



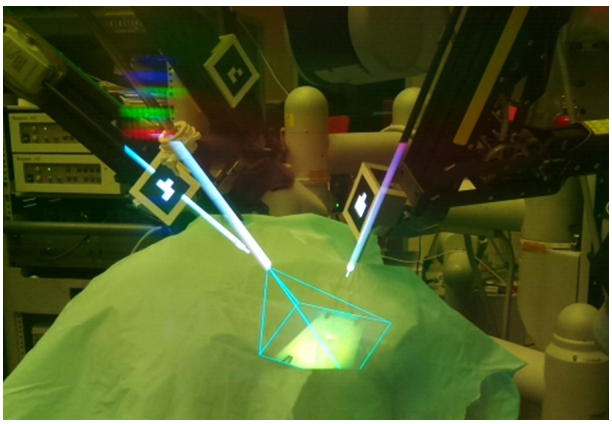
(c) After display calibration



(d) Overlay with hand-held instrument



(e) Virtual Monitor Visualization



(f) Frustum Projection Visualization

Figure 5.3: Visualization results of *ARssist*

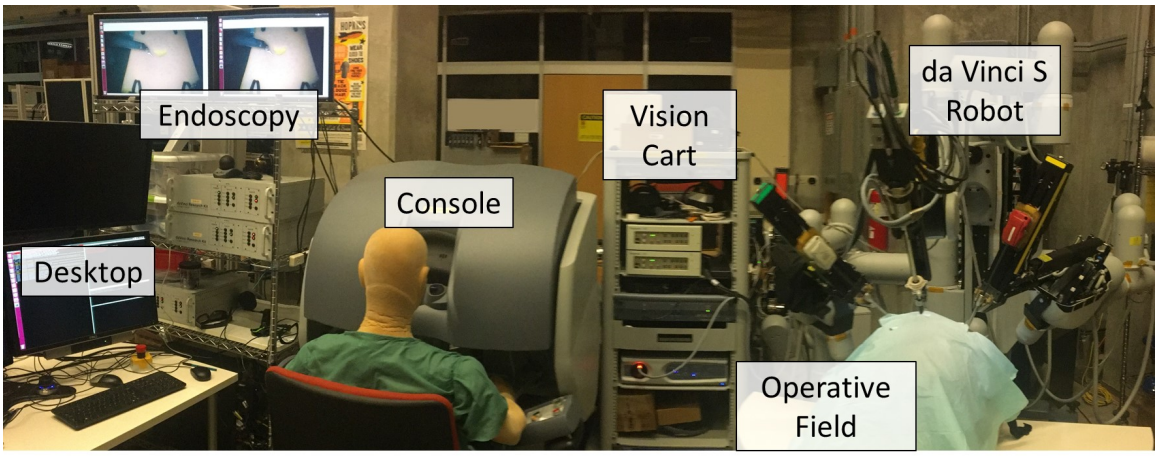


Figure 5.4: System setup of *ARssist*

doscopy at the end of the endoscope frustum, which is able to inform the FA not only about the endoscopy video itself, but also the geometry of the endoscope. Since the endoscope is also held by a robotic arm, *ARssist* obtains the kinematics of the endoscopic arm and calculates the pose of the endoscope at runtime. With a standard camera calibration of the endoscope, we calculate the horizontal and vertical field-of-view of the endoscope. Combining the pose and FOV, *ARssist* renders a frustum extending the tip of the endoscope and projects the endoscopy on a clipping plane of the frustum. The visualization result is shown in Fig. 5.3f. In this way, the disorientation issue of traditional laparoscopic surgery [286] is solved because the endoscopy is displayed in the correct orientation with respect to the world coordinate system.

5.4 System Implementation

Here we report the basic implementation of *ARssist*. During the several evaluation iterations of systems, we have refined our implementation based on user feedback, which will be discussed along the evaluation sections. We chose da Vinci as the robotic platform and Microsoft HoloLens as the OST-HMD.

5.4.1 Data Flow in *ARssist*

The data flow in *ARssist* is illustrated in Fig. 5.5. Orange boxes show the data that are obtained at runtime and are updated frequently. Blue boxes identify the data that are known prior to an instance of the application. Calibration data and the robot model (e.g., DH parameters, meshes) are considered prior knowledge. Green boxes are the visualization results and are the destinations of the data flow.

Both channels of the endoscopy are available through frame grabber at $30Hz$, with a resolution of 1920×1080 pixels. A computer program fetches the two channels of endoscopy, downscales the original images to 640×360, concatenates both channels, and streams it to the HoloLens via Motion-JPEG protocol.

A Unity application runs on the HoloLens as part of *ARssist*. Fiducial marker tracking is implemented based on HololensARToolKit (Sect. 2.6.8). The front-facing camera of HoloLens is configured for a resolution of 1344×768, 67° FOV and 15 fps. We use the robotic model provided by [69]. The rendering, socket communication and

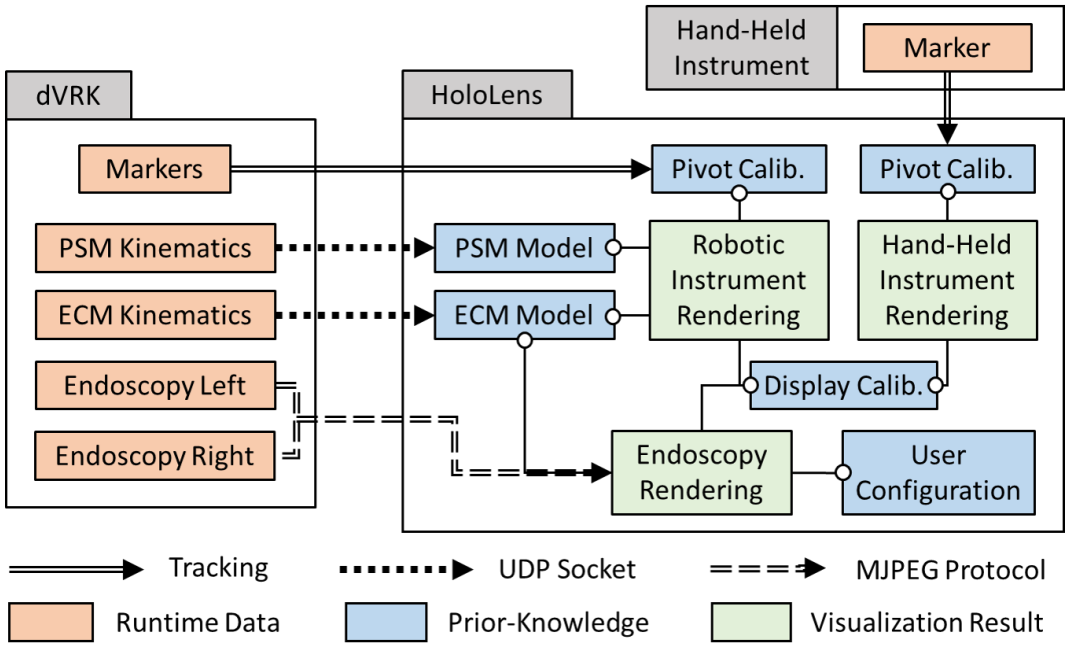


Figure 5.5: The Data Flow in *ARssist*

tracking are handled with different threads on HoloLens. We measure the frame rate for rendering, tracking, and endoscopy, which are $32.91 \pm 1.96\,Hz$, $13.64 \pm 0.78\,Hz$, and $26.57 \pm 3.10\,Hz$, respectively. The end-to-end latency of stereo endoscopy streaming is $220.81 \pm 25.54\,ms$, with a down-scaled image of 640×480 pixel resolution.

5.4.2 Sample Visualization of *ARssist*

Fig. 5.3 shows the visualization results captured by a pair of eye-simulating cameras placed behind the HoloLens.

We adapted the display calibration detailed in Sect. 2.6 to ensure the visualization is accurately overlaid for the user's vision. The average reprojection error for the display calibration is $4.27mm$ with standard deviation of $3.09mm$ (Fig. 5.3c).

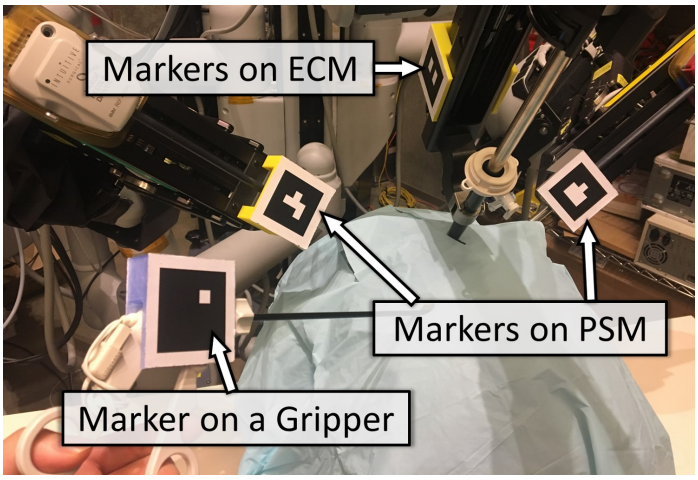


Figure 5.6: Fiducial markers on robotic arms and hand-held instrument

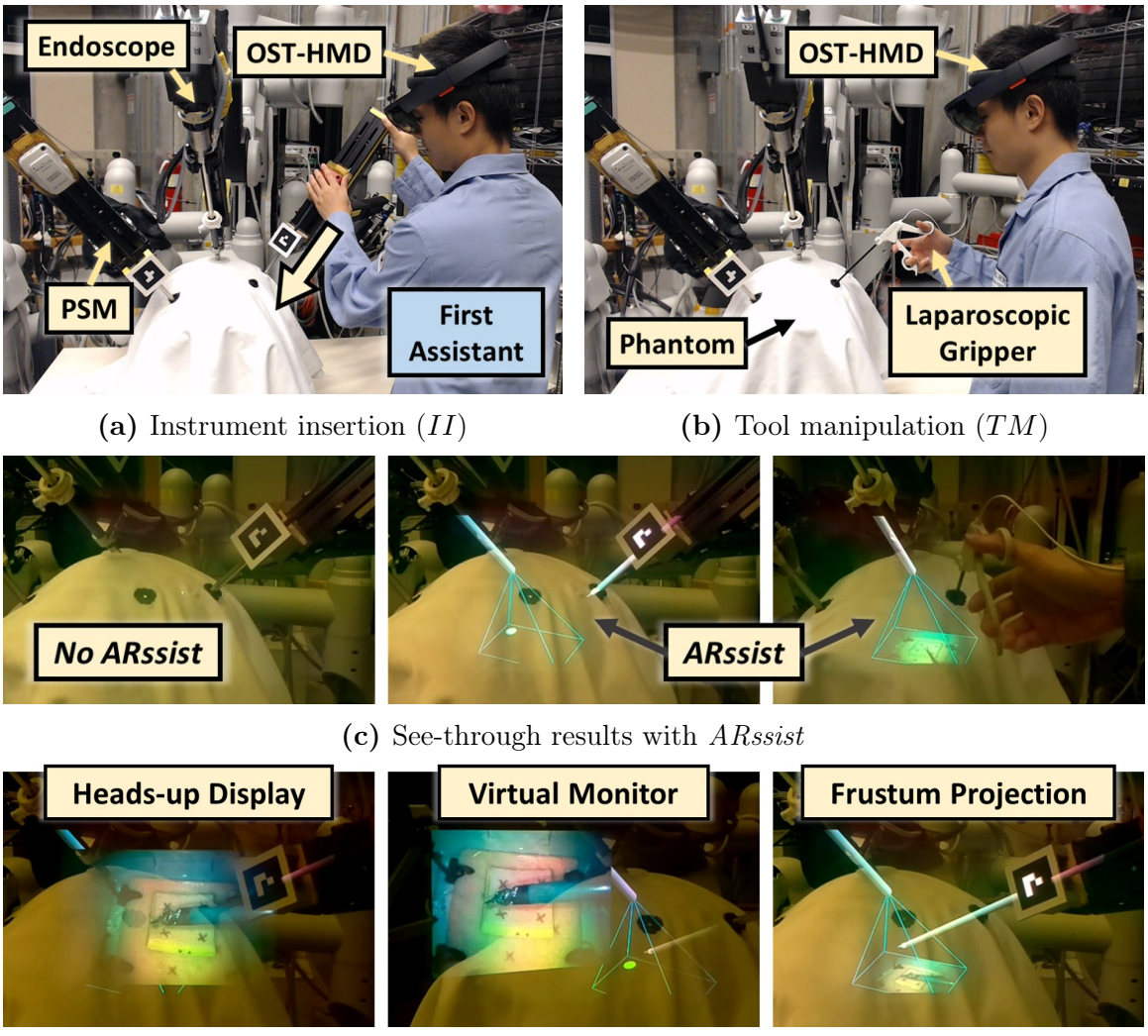
Fig. 5.3e and Fig. 5.3f demonstrate the different configurations for visualization of the stereo endoscopy. In Fig. 5.3e, the endoscopy is displayed on a world-anchored virtual monitor. In this way, the FA is able to see both the surgical field and the endoscopy with much less effort in terms of head rotation. Fig. 5.3f depicts the result for the frustum projection visualization. The viewing frustum of the endoscope is rendered at the tip of the ECM. The vertical and horizontal field-of-view of the endoscope are calculated from the endoscope camera calibration.

5.4.3 Voice Commands

We implemented a voice-based user interface to control the behavior of *ARssist*. This allows the user to select the desired endoscopy display by saying "*heads-up*" (head-anchored visualization),"*virtual monitor*" (world-anchored visualization), or "*frustum*" (frustum projection). The word "*next*" can iterate over the three methods.

Voice commands "*move back*", "*move forward*", "*larger*", and "*smaller*" can adjust the position and scale of the virtual monitor.

5.5 Tasks of the First Assistant



(a) Instrument insertion (II)

(b) Tool manipulation (TM)

(c) See-through results with *ARssist*

(d) Different visualization methods of endoscopy in *ARssist*

Figure 5.7: Instrument insertion and tool manipulation with and without *ARssist*

The involvement of the FA can be categorized based on the three phases of the

surgery. Before the surgery, the FA usually works with the surgeon for trocar placement and docking of the robot [134]. During the surgeon's operation, the FA is responsible for instrument exchange, manipulating laparoscopic tools depending on the specific procedure, and other tasks [134]. After the operation, the FA removes the instruments, undocks the robot, and performs port closure [8].

Instrument insertion (II) and tool manipulation (TM) are relatively frequent and repetitive tasks for the FA.

5.5.1 Instrument Insertion (II)

II occurs after the robot is docked, and when a robotic instrument needs to be changed due to a procedure requirement. TM refers to the maneuver of normal laparoscopic tools. For example, when the surgeon performs suturing, the FA must hold the suture with a laparoscopic gripper and pass the suture to the robotic instrument controlled by the surgeon.

5.5.2 Tool Manipulation (TM)

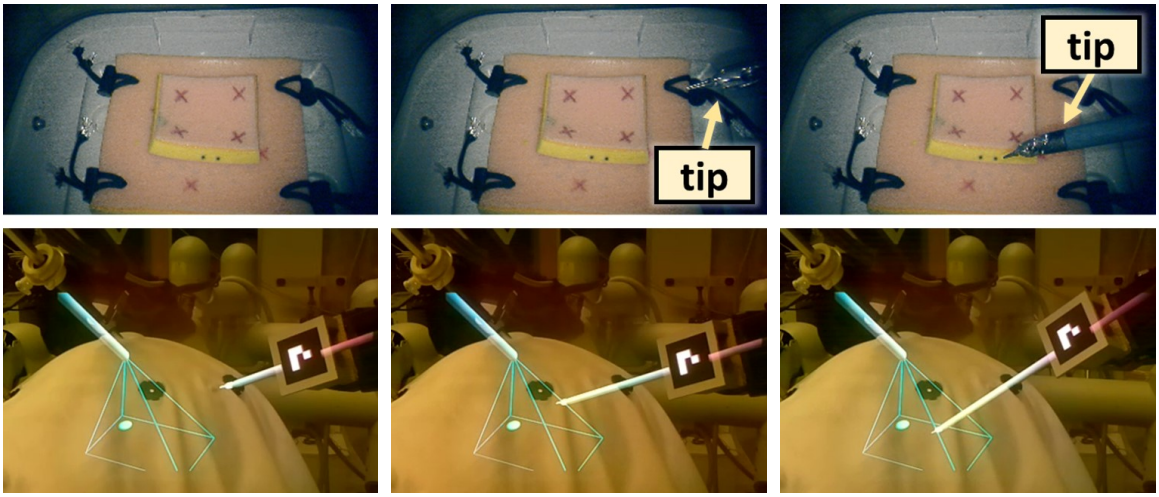
TM is also required for retraction, suction, and specimen extraction [229]. In the current surgical workflow, the endoscopy is displayed on a stationary monitor that is located far from the operation site and provides the only feedback to the FA. In II, the FA has to blindly navigate new instruments into the endoscopy, or ask the

surgeon to drive the endoscope to look at the inserted instrument to ensure safety. For TM, the FA also watches the external monitor for guidance and, depending on the location and orientation of the endoscope, the operation may create an awkward situation for the FA's hand-eye coordination [286].

5.6 Evaluation of *ARssist*

We evaluate instrument insertion (II) and tool manipulation (TM), both with (AR) and without (NA) the aid of *ARssist*, forming four scenarios: II_{NA}, II_{AR}, TM_{NA}, TM_{AR}.

5.6.1 Instrument Insertion: Procedure and Metric



(a) Before insertion **(b)** During insertion **(c)** Finish insertion

Figure 5.8: II_{AR}: instrument insertion with the help of *ARssist*

For II_{NA} and II_{AR}, the users are asked to insert the robotic instrument attached on the Patient-Side Manipulator (PSM) into the endoscopy's right side, assuming that the surgeon intends to control it using the right Master Tool Manipulator (MTMR). In II_{AR}, the user has the additional view of the AR content "within" the patient body, as shown in Fig. 5.8.

For data analysis, we manually annotate the time that the user starts insertion t_S, and the time when the insertion completes t_E based on the PSM kinematics data. We also annotate the time when the instrument tip appears in the endoscopy video, t_M. We compute the trajectory of the instrument tip (Fig. 5.9) based on the kinematics data and the Denavit-Hartenberg parameters of the PSM: $Q(t)$, $t_S \leq t \leq t_E$.

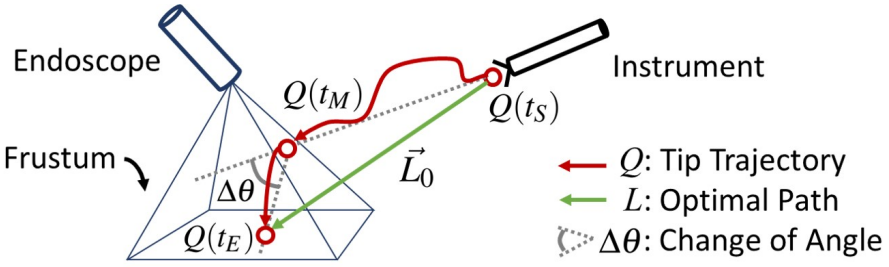


Figure 5.9: Subjective metric for evaluation of II

We define three objective metrics to evaluate user performance: navigation time (t_{Nav}), change of angle ($\Delta\theta$) and root-mean-square (RMS) distance of the trajectory (d_{RMS}).

5.6.1.1 Navigation Time t_{Nav}

Before the instrument appears in the endoscopy, the user is navigating the instrument with the guidance of *ARssist* (II_{AR}) or without any guidance (II_{NA}). This amount of time can be calculated as: $t_{Nav} = t_M - t_S$. If the user wrongly orients the instrument, the insertion will not be successful, and therefore, the user needs to spend more time to find out "where is the instrument", leading to longer navigation time. t_{Nav} determines the efficiency of the user.

5.6.1.2 Change of Angle $\Delta\theta$

We define $\Delta\theta$ as:

$$\begin{gathered} \vec{L_1} = Q(t_M) - Q(t_S), \quad \vec{L_2} = Q(t_E) - Q(t_M) \\ \Delta\theta = arccos(\vec{L_1} \cdot \vec{L_2})/(\|\vec{L_1}\| \cdot \|\vec{L_2}\|) \end{gathered} \tag{5.4}$$

The vector $\vec{L_1}$ represents the user's intended direction to insert the instrument before it appears in the endoscopic video. In case of II_{AR}, this intended direction is guided by the virtual instruments and FOV of endoscope rendered inside the phantom (Fig. 5.8), but in case of II_{NA}, it is based on the user's "feeling" or experience. When the instrument appears in the endoscopy (after t_M), the motion of the instrument will change if the user realizes that the instrument's path diverts from his/her intended path. $\vec{L_2}$ represents the subsequent corrected direction of motion. Therefore, the change of angle $\Delta\theta$ is defined as the angle between $\vec{L_1}$ and $\vec{L_2}$, as an indicator of the consistency of the insertion path (illustrated in Fig. 5.9).

5.6.1.3 Root-Mean-Square (RMS) Distance d_{RMS}

We treat the line between $Q(t_S)$ and $Q(t_E)$ as the optimal trajectory $\vec{L_0}$ (see Fig. 5.9). During the insertion, for each point on the user's trajectory $Q(t)$, we compute its distance to $\vec{L_0}$, and then calculate the RMS value:

$$\begin{aligned} d(t) &= DistancePointToLine(Q(t),\ \vec{L_0}) \\ d_{RMS} &= \sqrt{\frac{1}{N}\sum_{t_S<t<t_E}\|d(t)\|^2} \end{aligned} \tag{5.5}$$

where N is the total number of sample points on the trajectory. d_{RMS} represents the extent that the instrument tip diverts from the optimal path. Larger d_{RMS} means that the instrument tip moves further away from the desired path, which increases the potential of collision with an organ or other tissue. Thus, d_{RMS} is an indicator of operation safety.

5.6.2 Tool Manipulation: Procedure and Metric

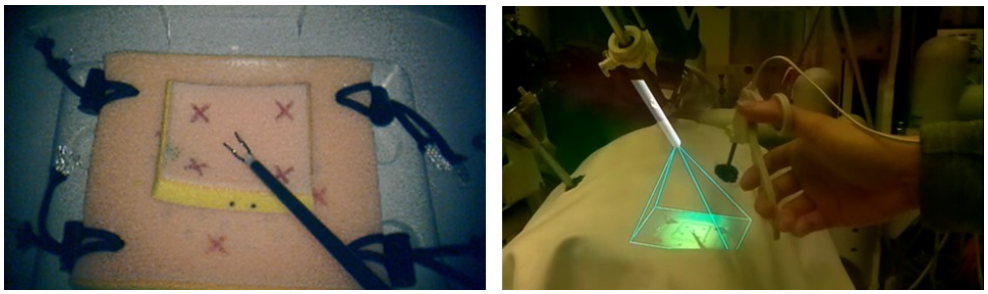


Figure 5.10: TM_{AR}: tool manipulation with the help of *ARssist*

For TM_{NA} and TM_{AR}, the users manipulate a laparoscopic gripper to retract one or more rubber rings out of the body. The rubber rings are visible in the endoscopy.

During the procedure, the user needs to first navigate the gripper into the FOV of endoscopy, then pick up the ring based on the feedback of endoscopy, and finally retract it from the site. In TM_{AR}, *ARssist* offers flexible configurations to display the endoscopic video (Fig. 5.10), while in TM_{NA}, the user can only watch the endoscopy on a monitor.

We annotate the time that the hand-held gripper first appears in the endoscopic video, t_S, and the time that it leaves the scene, t_E, and compute the following objective metric.

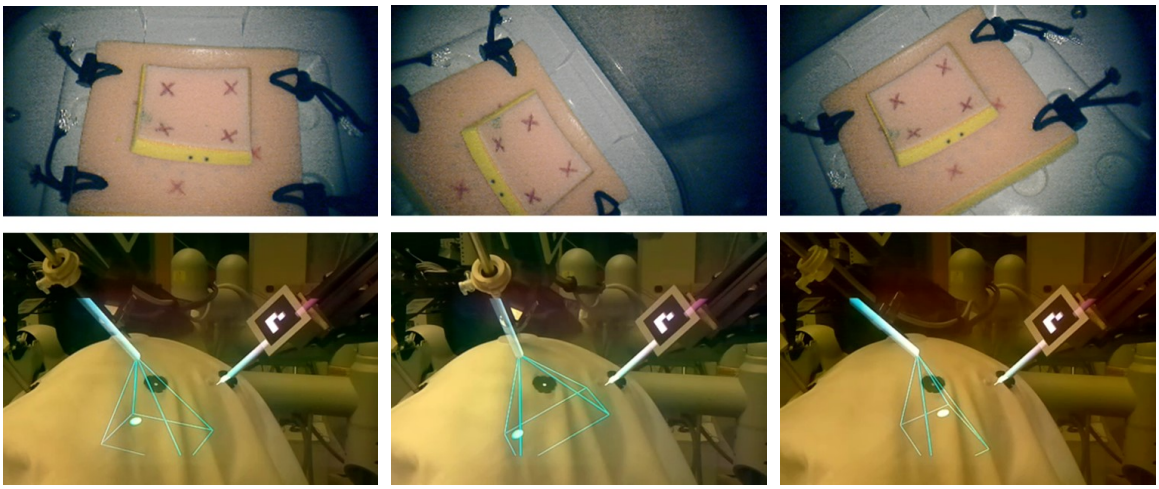
5.6.2.1 Manipulation Time t_{Mani}

The manipulation time t_{Mani} is defined by $t_E - t_S$, as the total amount of time that the user is guided by the endoscopic video to grab and retract the rubber ring. We do not consider the time spent on inserting the gripper into the FOV of endoscope, which is mainly addressed in the task of instrument insertion.

5.6.3 Pose of Endoscope

We use a straight endoscope for the pilot run and a 30°-angled endoscope for the multi-user study, both manufactured by Intuitive Surgical, Inc. For each of the four scenarios, we position the endoscope programmatically at three different poses: EP_1, EP_2 and EP_3 (Fig. 5.11). Therefore, there are 12 trials in total for each user:

$$II_{NA}^{EPn}\ ,\ II_{AR}^{EPn}\ ,\ TM_{NA}^{EPn}\ ,\ TM_{AR}^{EPn},\ n = 1, 2, 3 \tag{5.6}$$



(a) EP_1 **(b)** EP_2 **(c)** EP_3

Figure 5.11: Different poses of the endoscope for the experiment

Among the three poses, EP_1 is the most normal configuration, where the horizontal axis of the endoscopic video is parallel to the transverse axis of the phantom. EP_2 and EP_3 are two less common, but still clinically possible, endoscope setups. In EP_2, the endoscope is oriented so that it and the user have a similar perspective, whereas EP_3 is the most awkward setup for the user's hand-eye coordination.

5.6.4 Experimental Procedure

The experimental procedure for both the pilot run and user study is as follows:

1. Complete the consent form (user study only) and pre-experiment survey
2. Training of II & TM, with and without $ARssist$, repeated multiple times, with multiple endoscope poses
3. Perform II_{NA}^{EPn} & II_{AR}^{EPn}, $n = 1, 2, 3$ in randomized order

4. Complete the post-experiment survey for II
5. Perform TM_{NA}^{EPn} & TM_{AR}^{EPn}, $n = 1, 2, 3$ in randomized order (2 rings for pilot run, 1 ring for user study)
6. Complete the post-experiment survey for TM
7. Conduct an informal interview (pilot run only)

The post-experiment questionnaire includes: *i*) self-reported ratings of *outcome*, *speed*, *confidence*, *satisfaction*, *fatigue*, *interest* and *hand-eye coordination* for performance with and without *ARssist* ($0 \sim 5$), *ii*) preference of the three endoscopy visualization methods ($0 \sim 5$), and *iii*) whether *FOV*, *smoothness*, *latency*, *accuracy* of the virtual overlay or any other factor is limiting the current application (Y/N).

We record all related data for each trial, including the robot kinematics, tracking results, stereo endoscopic video, and questionnaire results. All data are timestamped with millisecond accuracy.

5.7 Pilot Run and Interviews with Surgeons

We invited three surgeons who frequently perform surgeries with the da Vinci robot for the pilot study and interview. All of them have experience working as the first assistant. Their background information is listed in Tab. 5.2. The setup for pilot run is shown in Fig. 5.12.

The surgeons followed the experimental procedure outlined in Sec. 5.6.4. Their

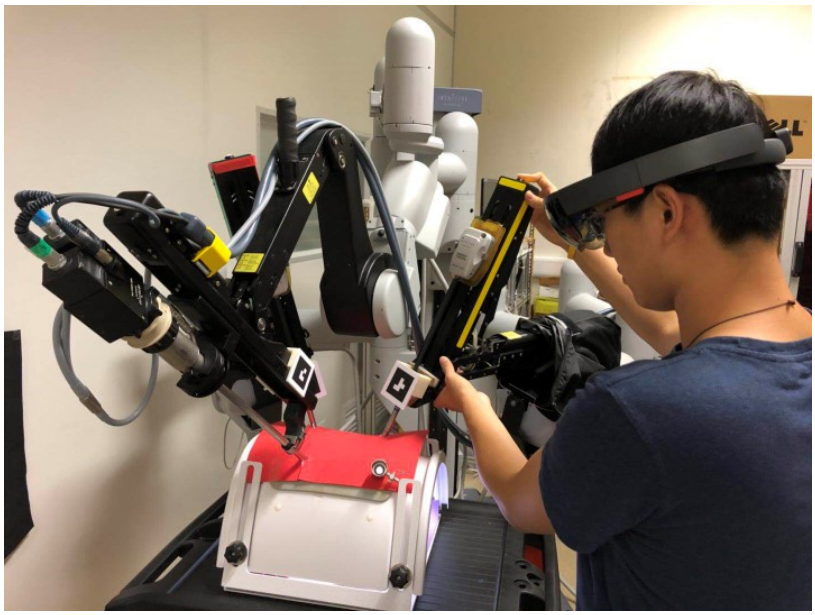


Figure 5.12: *ARssist* setup at CUHK for pilot study

Table 5.2: Background information for the invited surgeons

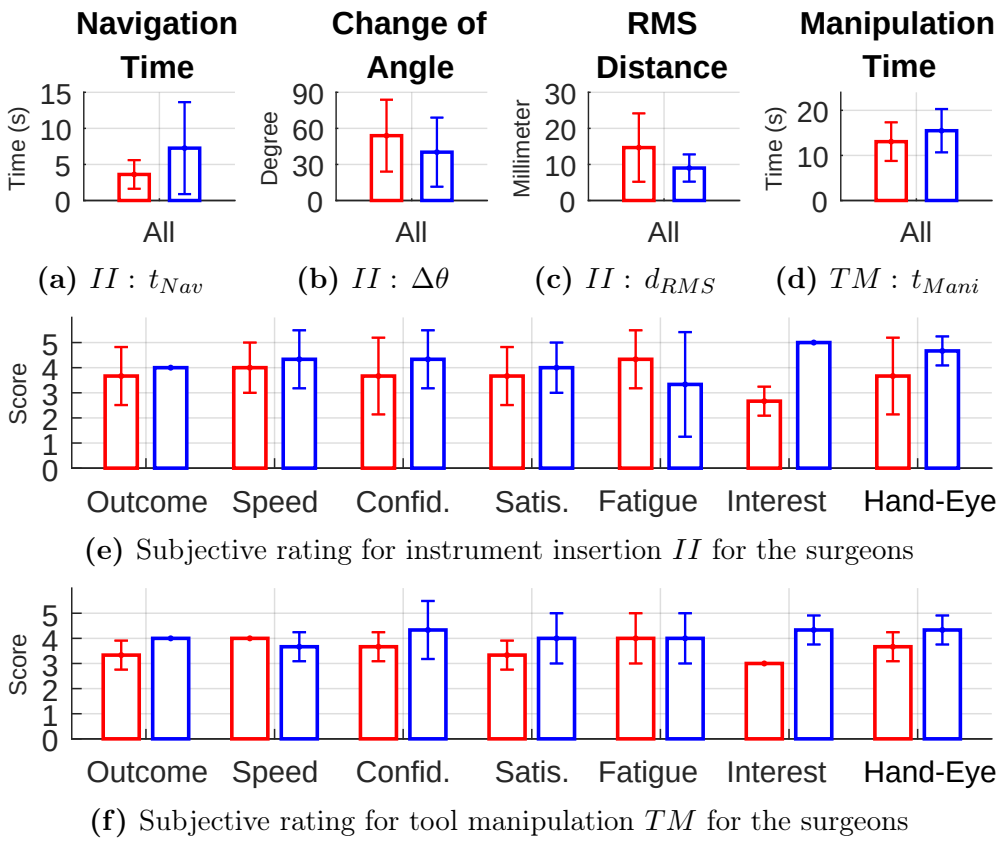
ID	Age	Gender	Most practiced surgeries	Frequency
#1	39	Female	Hepatectomy	2-4/mo.
#2	36	Male	Radical Prostatectomy	3-4/mo.
#3	33	Male	Prostatectomy, crystectomy	3-4/mo.

performance data and subjective feedback are shown in Fig. 5.13.

5.7.1 Results and Discussion

5.7.1.1 Instrument Insertion

Figs. 5.13a, 5.13b and 5.13c show the results of the objective metrics. The navigation time t_{Nav} for these experienced users is $7.26 \pm 6.37\,s$ in II_{AR}, which is longer than their normally practised condition ($3.61 \pm 1.92\,s$). The trajectory profile indicates im-



(a) $II : t_{Nav}$ **(b)** $II : \Delta\theta$ **(c)** $II : d_{RMS}$ **(d)** $TM : t_{Mani}$

(e) Subjective rating for instrument insertion II for the surgeons

(f) Subjective rating for tool manipulation TM for the surgeons

Figure 5.13: Results for the pilot run with the surgeons ($N = 3$). Red: without *ARssist*. Blue: with *ARssist*.

provement using *ARssist*. The average change of angle $\Delta\theta$ is reduced by 25.44% from $53.95° \pm 29.90°$ in II_{NA} to $40.22° \pm 28.74°$ in II_{AR}. The RMS distance d_{RMS} of the trajectory is reduced by 38.64% on average, from $14.68 \pm 9.48\,mm$ for II_{NA} to $9.01 \pm 3.77\,mm$ for II_{AR}. However, with the limited number of samples (N=3), the results cannot determine any significant difference between the performance.

5.7.1.2 Tool Manipulation

Fig. 5.13d shows the average time for the surgeons to manipulate the gripper to retract a rubber ring from the phantom in TM_{NA} ($13.06 \pm 4.28\,s$) and TM_{AR} ($15.48 \pm 4.81\,s$). It is still slightly easier for the surgeons to manipulate under the guidance of endoscopy displayed on a traditional monitor, partially due to the fact that laparoscopic surgeons are especially trained for instrument maneuver under challenging hand-eye coordination conditions [55].

5.7.1.3 Subjective Feedback

Figs. 5.13e and 5.13f show the subjective ratings of the surgeons for II and TM, which indicate no substantial preference between the AR and non-AR cases. In general, the surgeons do find the task with *ARssist* to be more interesting than without AR guidance.

5.7.1.4 Interview Results

Surgeon #1 performs liver resection more frequently. Her FA is usually in charge of changing instruments, applying clips/staples, and passing needles. She pointed out some major limitations of the current implementation: i) the HoloLens is too heavy to wear for a long time, ii) there is perceivable lag for the virtual overlay, especially for the hand-held gripper, and iii) the FA needs some training to use the AR system. She agreed that if the limitations are addressed, *ARssist* can be integrated

into hepatectomy and will be useful for the FA.

Surgeon #2 performs robotic radical prostatectomy and also occasionally works as a FA. His FA helps bring in instruments, apply suction/retraction and pass in clips and needles. He thinks that *ARssist* can be beneficial for the FA: i) during initial insertion of an instrument at the start of the surgery, ii) during instrument insertion in the middle of surgery, and iii) especially when an angled-endoscope is used. The major limitation is the small FOV of HoloLens. He believes that *ARssist* can mainly benefit a less experienced FA in terms of operation time and safety.

Surgeon #3 performs robotic radical prostatectomy, robotic radical crystectomy and related procedures. During his operation, the FA helps in traction, passing sutures, and clipping vessels. He thinks *ARssist* can improve the performance of the FA: i) in initial setup of the robotic instruments, and ii) facilitate the insertion of the hand-held instruments for passing suture and clipping vessels. The resolution of HoloLens is a current limitation. He believes that the system will be especially useful for an inexperienced FA.

5.7.2 Summary

In the pilot run with the surgeons, *ARssist* extended the navigation time in II and the manipulation time in retracting the rubber ring, but improved the path consistency and operation safety in II. From the subjective ratings, there is not much significant difference between using and not using AR guidance. In the subsequent

interview, all surgeons agreed that the FA plays an important role in robotic-assisted surgeries and that our system can be beneficial for the FA, especially for an inexperienced FA. If the current limitations are solved, *ARssist* will be useful for their current tasks, and therefore would improve the overall quality of the surgery.

5.8 User Study at Johns Hopkins University

Based on the feedback of pilot run, we removed the tracking and overlay of hand-held tools due to frequent failure. We added the support for a 30°-angled endoscope in addition to the straight endoscope (Fig. 5.14).

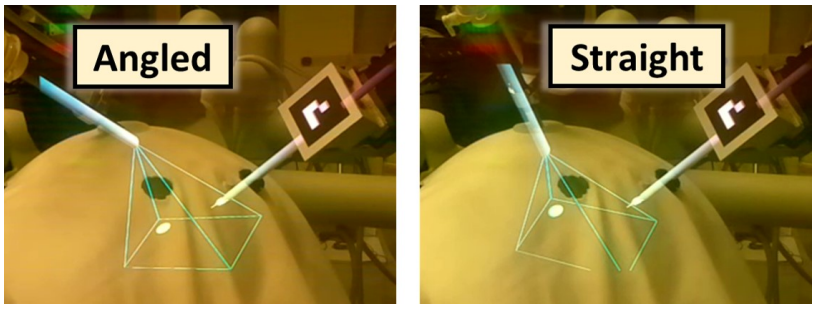
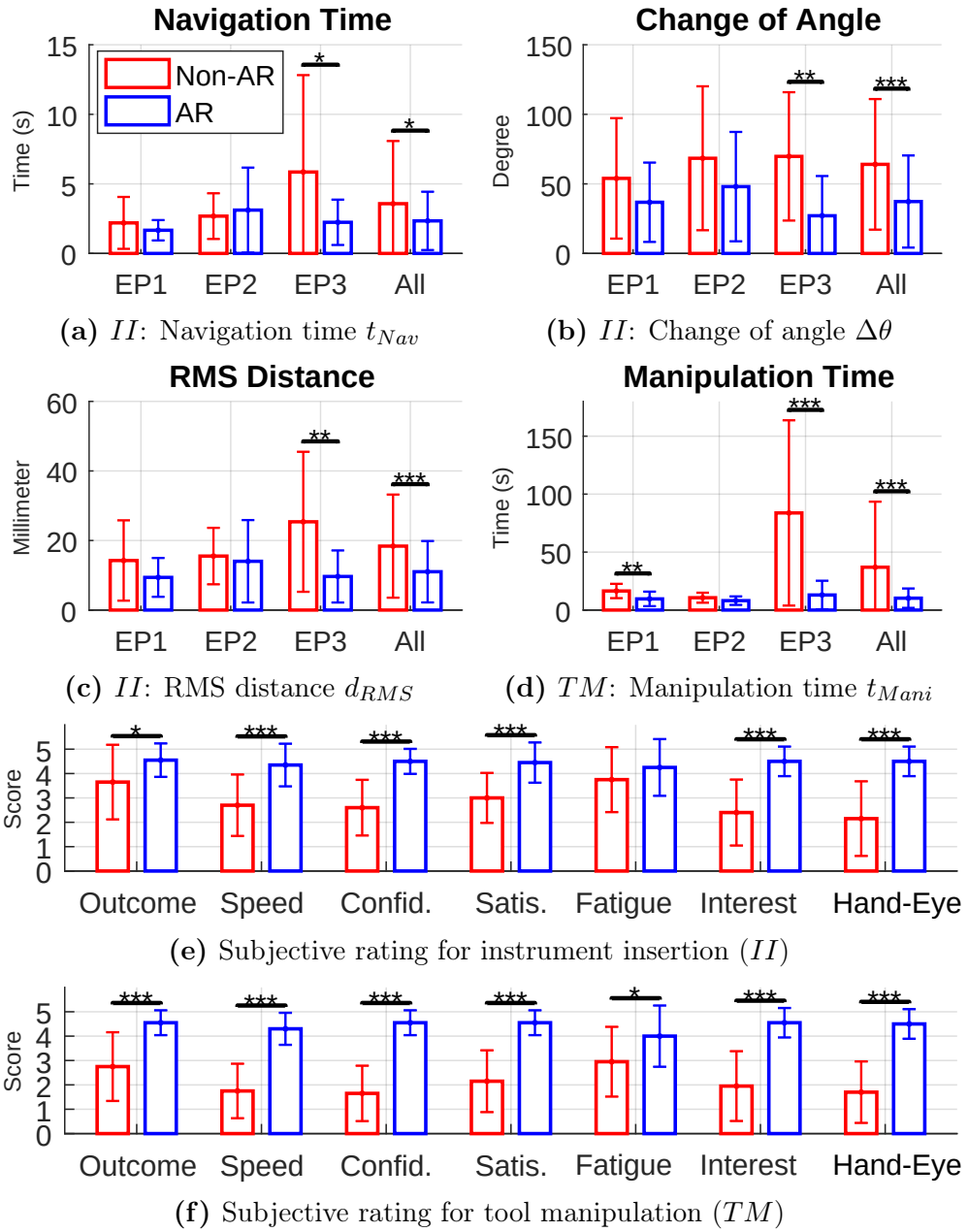


Figure 5.14: 30°-angled and straight endoscope for *ARssist*

After refining the experiment setup, we conducted a multi-user study (HIRB 00007467) with 20 inexperienced subjects (gender: 17 male, 3 female; age: mean 26.65, std 8.00). The rationale for using inexperienced (non-medical) subjects is that the goal of *ARssist* is to improve spatial awareness and hand-eye coordination, rather than to provide medical information, and therefore medical knowledge is not a prerequisite.

5.8.1 Results and Discussion

The results are shown in Fig. 5.15, Tab. 5.3, Tab. 5.4 and Tab. 5.5.



(a) II: Navigation time t_{Nav}

(b) II: Change of angle $\Delta\theta$

(c) II: RMS distance d_{RMS}

(d) TM: Manipulation time t_{Mani}

(e) Subjective rating for instrument insertion (II)

(f) Subjective rating for tool manipulation (TM)

Figure 5.15: Results for the user study at Johns Hopkins University with inexperienced users ($N = 20$). Red: without *ARssist*. Blue: with *ARssist*.

Table 5.3: Results for the user study at Johns Hopkins University with inexperienced users ($N = 20$), corresponding to Fig. 5.15a to Fig. 5.15d. Data is represented as mean ± standard deviation. Better values are highlighted in bold font.

Metric	Port	Non-AR	*ARssist*	p-value
Navigation Time (s)	EP_1	2.19 ± 1.87	$\mathbf{1.67 \pm 0.73}$	0.25
	EP_2	$\mathbf{2.68 \pm 1.64}$	3.12 ± 3.05	0.58
	EP_3	5.85 ± 6.96	$\mathbf{2.24 \pm 1.63}$	$\mathbf{2.96 \times 10^{-2}}$
	All	3.58 ± 4.50	$\mathbf{2.34 \pm 2.10}$	$\mathbf{4.53 \times 10^{-2}}$
Change of Angle (°)	EP_1	53.96 ± 43.29	$\mathbf{36.79 \pm 28.52}$	0.15
	EP_2	68.41 ± 51.73	$\mathbf{48.04 \pm 39.30}$	0.17
	EP_3	69.76 ± 46.13	$\mathbf{27.10 \pm 28.55}$	$\mathbf{1.15 \times 10^{-3}}$
	All	64.06 ± 46.93	$\mathbf{37.31 \pm 33.11}$	$\mathbf{4.51 \times 10^{-4}}$
RMS Distance (mm)	EP_1	14.25 ± 11.55	$\mathbf{9.39 \pm 5.58}$	9.85×10^{-2}
	EP_2	$15.52 \pm \mathbf{8.11}$	$\mathbf{14.02} \pm 11.85$	0.64
	EP_3	25.38 ± 20.13	$\mathbf{9.66 \pm 7.49}$	$\mathbf{2.26 \times 10^{-3}}$
	All	18.38 ± 14.83	$\mathbf{11.02 \pm 8.83}$	$\mathbf{8.59 \times 10^{-4}}$
Manipulation Time (s)	EP_1	$16.46 \pm \mathbf{6.17}$	$\mathbf{9.64} \pm 6.26$	$\mathbf{1.32 \times 10^{-3}}$
	EP_2	10.65 ± 4.32	$\mathbf{8.13 \pm 3.68}$	5.42×10^{-2}
	EP_3	83.78 ± 79.91	$\mathbf{13.00 \pm 12.30}$	$\mathbf{3.63 \times 10^{-4}}$
	All	36.96 ± 56.52	$\mathbf{10.26 \pm 8.36}$	$\mathbf{2.50 \times 10^{-5}}$

5.8.1.1 Instrument Insertion

Fig. 5.15a shows that on average, *ARssist* reduced the navigation time t_{Nav} by 34.57%, from $3.58 \pm 4.50\,s$ in II_{NA} to $2.34 \pm 2.10\,s$ in II_{AR}. For each endoscope pose EP_i, we use a t-test to determine whether the difference between II_{AR} and II_{NA} is significant. Here we assume that the data follow normal distribution. For EP_3, the users spent $2.24 \pm 1.63\,s$ to navigate in II_{AR}, which is significantly ($p = 2.96 \times 10^{-2}$)

shorter than the amount of time spent in II_{NA}: $5.85 \pm 6.96\,s$. EP_1 and EP_2 do not show significance with a t-test. The user's navigation time is dependent on two categorical variables: i) II_{AR} or II_{NA}, and ii) pose of endoscope: EP_1, EP_2 or EP_3. Therefore, we use a two-way ANOVA to test whether t_{Nav} is significantly affected by the two variables. The result shows that both factors are significant for t_{Nav} ($p_{AR} = 4.53 \times 10^{-2}, p_{EP} = 2.09 \times 10^{-2}$).

As seen in Fig. 5.15b, the change of angle $\Delta\theta$ is reduced by 41.74% from $64.04 \pm 46.93°$ for II_{NA} to $37.31 \pm 33.11°$ in II_{AR}. Within all groups of endoscope pose, the average $\Delta\theta$ is smaller in II_{AR}. With a t-test, only EP_3 shows significant improvement ($p = 1.15 \times 10^{-3}$). A two-way ANOVA test of $\Delta\theta$ shows that the use of *ARssist* significantly affects user performance ($p_{AR} = 4.51 \times 10^{-4}$).

The results of RMS distance d_{RMS} are shown in Fig. 5.15c. The mean d_{RMS} is reduced by 40.04%, from $18.39 \pm 14.83\,mm$ in II_{NA} to $11.02 \pm 8.83\,mm$ in II_{AR}. This reduction is achieved for all tested endoscope poses EP_1, EP_2 and EP_3, but is only significant for EP_3 ($p = 2.26 \times 10^{-3}$). The two-way ANOVA shows that, with *ARssist*, the reduction of d_{RMS} is quite significant ($p_{AR} = 8.59 \times 10^{-4}$).

We extract the participants' subjective ratings for their experience with and without *ARssist* from the questionnaire. The detailed results are shown in Tab. 5.4. We assume that the subjective ratings in each category and each task condition follow normal distribution. The t-test shows that users significantly prefer II_{AR} in *outcome* ($p = 2.15 \times 10^{-2}$), *speed* ($p = 2.41 \times 10^{-5}$), *confidence* ($p = 4.82 \times 10^{-8}$),

Table 5.4: Subjective rating results of instrument insertion for the user study at Johns Hopkins University with inexperienced users ($N = 20$), corresponding to Fig. 5.15e. Data is represented as mean ± standard deviation. Better results are highlighted in bold font. Higher rating for "Fatigue" corresponds to less fatigue.

Metric	Non-AR	*ARssist*	p-value
Outcome	3.65 ± 1.53	$\mathbf{4.55 \pm 0.69}$	$\mathbf{2.15 \times 10^{-2}}$
Speed	2.70 ± 1.26	$\mathbf{4.35 \pm 0.88}$	$\mathbf{2.41 \times 10^{-5}}$
Confidence	2.60 ± 1.14	$\mathbf{4.50 \pm 0.51}$	$\mathbf{4.82 \times 10^{-8}}$
Satisfaction	3.00 ± 1.03	$\mathbf{4.45 \pm 0.83}$	$\mathbf{1.68 \times 10^{-5}}$
Fatigue	3.75 ± 1.33	$\mathbf{4.25 \pm 1.16}$	0.21
Interest	2.40 ± 1.35	$\mathbf{4.50 \pm 0.61}$	$\mathbf{2.00 \times 10^{-7}}$
Coordination	2.15 ± 1.53	$\mathbf{4.50 \pm 0.61}$	$\mathbf{1.71 \times 10^{-7}}$

satisfaction ($p = 1.68 \times 10^{-5}$), *interest* ($p = 2.00 \times 10^{-7}$) and *hand-eye coordination* ($p = 1.71 \times 10^{-7}$). In terms of *fatigue*, the average rating in II_{AR} is not significantly higher than in II_{NA} ($p = 0.21$). The results showed that the fatigue level was not significantly different with and without AR, because the weight of the OST-HMD offset some of the advantages of improved hand-eye coordination. All other subjective metrics heavily favor ARssist for instrument insertion task.

5.8.1.2 Tool Manipulation

The tool manipulation time t_{Mani} is shown in Fig. 5.15d and Tab. 5.3. It is $10.26 \pm 8.36\,s$ in TM_{AR}, which is significantly shorter than $36.96 \pm 56.52\,s$ in TM_{NA} (72.25%). The two-way ANOVA test shows that the reduction of t_{Mani} is significant ($p_{AR} = 2.50 \times 10^{-5}$). The average time reduction is also very significant for EP_1

($p = 1.32 \times 10^{-3}$) and EP_3 ($p = 3.63 \times 10^{-4}$) after t-test. Especially for EP_3 in TM_{NA}, many users spend a lot of time adjusting the gripper to approach the rubber ring under bad hand-eye coordination. With *ARssist*, the FOV of the endoscope is directly visualized for the user, therefore the spatial relationship between the gripper's motion and its appearance in the endoscopic video is naturally registered.

Table 5.5: Subjective rating results of tool manipulation for the user study at Johns Hopkins University with inexperienced users ($N = 20$), corresponding to Fig. 5.15f. Data is represented as mean ± standard deviation. Better results are highlighted in bold font. Higher rating for "Fatigue" corresponds to less fatigue.

Metric	Non-AR	*ARssist*	p-value
Outcome	2.75 ± 1.41	$\mathbf{4.55 \pm 0.51}$	$\mathbf{4.16 \times 10^{-6}}$
Speed	1.75 ± 1.12	$\mathbf{4.30 \pm 0.66}$	$\mathbf{1.07 \times 10^{-10}}$
Confidence	1.65 ± 1.14	$\mathbf{4.55 \pm 0.51}$	$\mathbf{1.11 \times 10^{-11}}$
Satisfaction	2.15 ± 1.27	$\mathbf{4.55 \pm 0.51}$	$\mathbf{1.78 \times 10^{-9}}$
Fatigue	2.95 ± 1.43	$\mathbf{4.00 \pm 1.26}$	$\mathbf{1.83 \times 10^{-2}}$
Interest	1.95 ± 1.43	$\mathbf{4.55 \pm 0.60}$	$\mathbf{5.55 \times 10^{-9}}$
Coordination	1.70 ± 1.26	$\mathbf{4.50 \pm 0.61}$	$\mathbf{6.80 \times 10^{-11}}$

The subjective results for tool manipulation are shown in Fig. 5.15f and Tab. 5.5. For TM_{NA}, the average ratings for *outcome*, *speed*, *confidence*, *satisfaction*, *fatigue*, *interest* and *hand-eye coordination* are 2.8, 1.8, 1.7, 2.2, 3.0, 2.0 and 1.7, respectively. For TM_{AR}, the average ratings are 4.6, 4.3, 4.6, 4.6, 4.0, 4.6 and 4.5. The improvement with *ARssist* is significant in the users' ratings in all listed metrics.

5.8.1.3 Preference for Endoscopy Visualization

As described in Sect. 5.3.4, *ARssist* provides three visualization methods to render the stereo endoscopic video. We extract the user's selection of visualization methods and their subjective ratings for them, as listed in Tab. 5.6.

Table 5.6: Total amount of time, number of selections, and user's rating for each visualization method of the stereo endoscopy ($N = 20$)

	Instrument Insertion			Tool Manipulation		
Method	Time	No.	Rating	Time	No.	Rating
Heads-up display	12.39 *s*	2	2.0	31.15 *s*	3	2.0
Virtual monitor	42.47 *s*	8	2.2	21.08 *s*	2	1.9
Frustum projection	223.77 *s*	54	4.5	552.87 *s*	58	4.9

Note that some users switched the visualization method during the operation, therefore, the total number of selections exceeds the number of trials (60). Most users selected the frustum projection method for both II and TM. This method orients the video at the correct geometric pose and restores the hand-eye coordination of the user, which is especially helpful for inexperienced users who have not been trained for hand-eye coordination.

5.8.2 Summary

The user study at Johns Hopkins University showed that *ARssist* can benefit inexperienced users by improving efficiency (34.57% shorter navigation time), navigation consistency (41.74% less change of angle), and safety (40.04% lower RMS path devia-

tion) for instrument insertion, and by enhancing hand-eye coordination (72.25% less time) in tool manipulation. It can be confirmed that inexperienced users performs significantly better with the assistance of *ARssist*.

5.9 User Study at Intuitive Surgical Inc.

There is no major refinement to the system before the user study taking place at Intuitive Surgical Inc. (Sunnyvale, CA) in October 2018, except for changing the dVRK system to the da Vinci Si system as the robot. Therefore, the system calibration is re-done. The stereo endoscopic video is captured using Blackmagic DeckLink Duo 2. The setup is shown in Fig. 5.16.

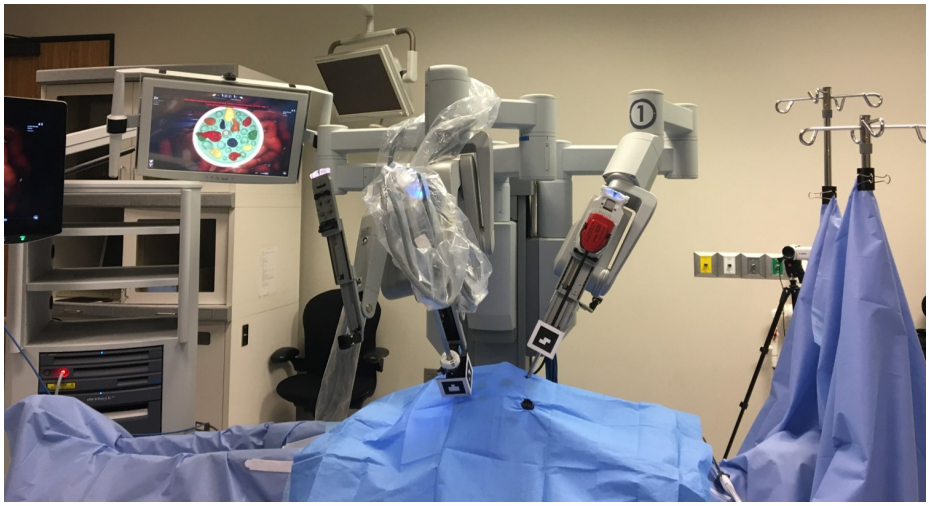


Figure 5.16: The experiment setup at Intuitive Surgical Inc.

During the study, 10 relatively experienced users were recruited, including robotic surgeons, clinical-development engineers and employees of the company that are fa-

miliar with laparoscopic procedures. The tasks, experiment procedure, and evaluation protocol are exactly the same as the user study at Johns Hopkins University. Additional third-view cameras were placed inside the mock operating room to capture the experiment procedure. The study was approved by the IRB at Intuitive Surgical Inc.

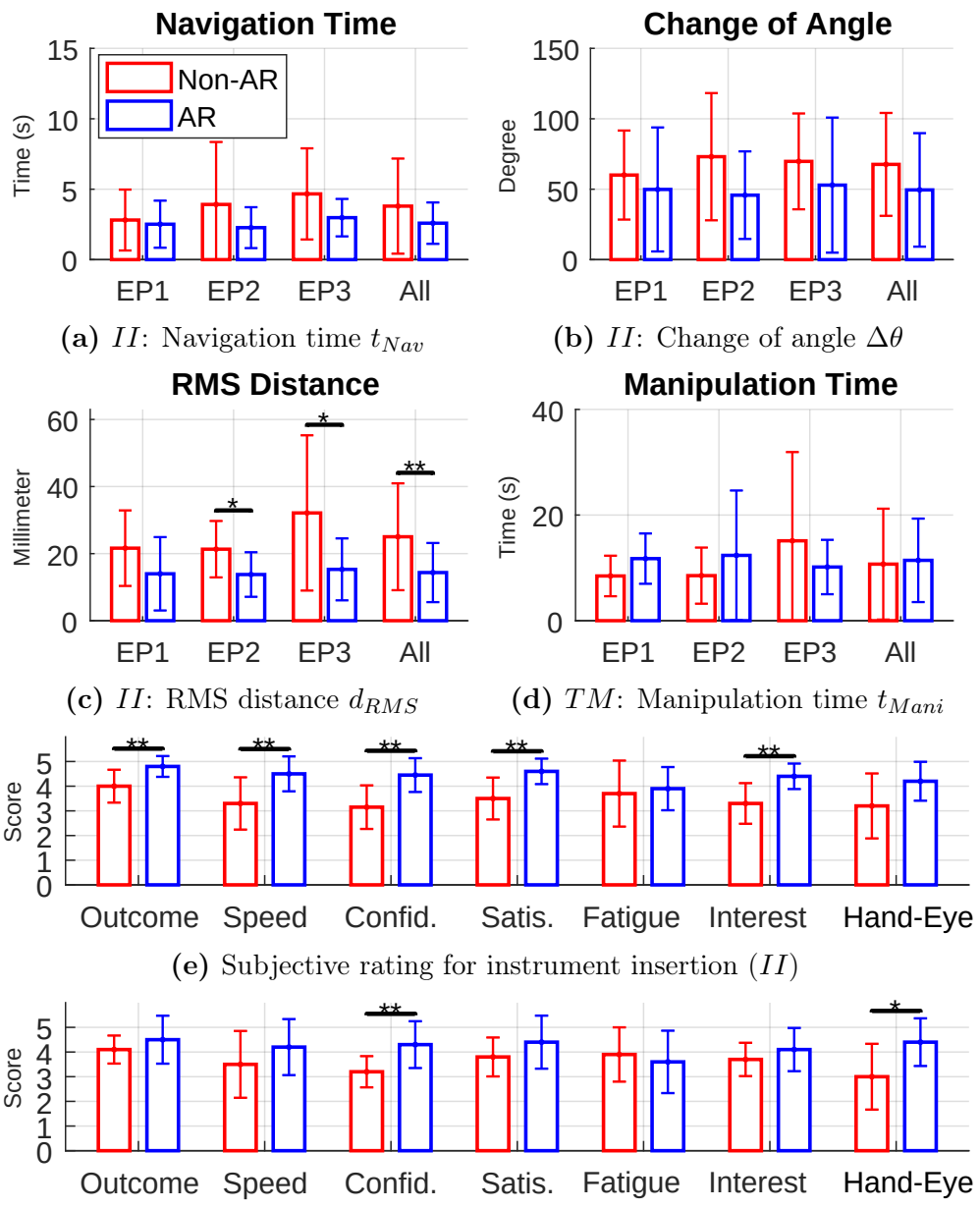
5.9.1 Results and Discussion

The results are shown in Fig. 5.17, Tab. 5.7, Tab. 5.8 and Tab. 5.9.

5.9.1.1 Instrument Insertion

Fig. 5.17a and Tab. 5.7 show that on average, *ARssist* reduced the navigation time t_{Nav} by 31.85%, from $3.80 \pm 3.38\,s$ in II_{NA} to $2.59 \pm 1.47\,s$ in II_{AR}. We assume the evaluation results for each metric in each condition follow normal distribution. For each endoscope pose EP_i, we use a t-test to determine whether the difference between II_{AR} and II_{NA} is significant. Although for all of the EP, the reduction in navigation time is shown, but in none of them the reduction achieves significance. The insufficient number of samples is a potential reason for the insignificance. We use a two-way ANOVA to test whether t_{Nav} is significantly affected by *ARssist*, with the same method in the evaluation presented in the previous section. The result shows that the effect is not significant for t_{Nav} ($p_{AR} = 8.03 \times 10^{-2}$).

As seen in Fig. 5.17b, the change of angle $\Delta\theta$ is reduced by 26.81% from $67.63 \pm 36.52^\circ$ for II_{NA} to $49.50 \pm 40.30^\circ$ in II_{AR}. Within all groups of endoscope pose, the



(a) II: Navigation time t_{Nav}

(b) II: Change of angle $\Delta\theta$

(c) II: RMS distance d_{RMS}

(d) TM: Manipulation time t_{Mani}

(e) Subjective rating for instrument insertion (II)

(f) Subjective rating for tool manipulation (TM)

Figure 5.17: Results for the study at Intuitive Surgical Inc. with experienced users ($N = 10$). Red: without *ARssist*. Blue: with *ARssist*.

average $\Delta\theta$ is smaller in II_{AR}. With a t-test, none of the EPs shows significant improvement. A two-way ANOVA test of $\Delta\theta$ shows that the use of *ARssist* does not

Table 5.7: Results for the user study at Intuitive Surgical Inc. with experienced users ($N = 10$), corresponding to Fig. 5.17a to Fig. 5.17d. Data is represented as mean ± standard deviation. Better values are highlighted in bold font.

Metric	Port	Non-AR	*ARssist*	p-value
Navigation Time (s)	EP_1	2.81 ± 2.16	$\mathbf{2.51 \pm 1.67}$	0.74
	EP_2	3.92 ± 4.43	$\mathbf{2.27 \pm 1.45}$	0.28
	EP_3	4.67 ± 3.24	$\mathbf{2.98 \pm 1.33}$	0.14
	All	3.80 ± 3.38	$\mathbf{2.59 \pm 1.47}$	8.03×10^{-2}
Change of Angle (°)	EP_1	$60.05 \pm \mathbf{31.60}$	$\mathbf{49.83} \pm 44.02$	0.56
	EP_2	73.09 ± 45.16	$\mathbf{45.75 \pm 31.10}$	0.13
	EP_3	$69.75 \pm \mathbf{34.03}$	$\mathbf{52.90} \pm 47.96$	0.38
	All	$67.63 \pm \mathbf{36.52}$	$\mathbf{49.50} \pm 40.30$	8.16×10^{-2}
RMS Distance (mm)	EP_1	21.62 ± 11.26	$\mathbf{13.99 \pm 10.94}$	0.14
	EP_2	21.32 ± 8.41	$\mathbf{13.76 \pm 6.64}$	$\mathbf{3.88 \times 10^{-2}}$
	EP_3	32.13 ± 23.15	$\mathbf{15.30 \pm 9.24}$	$\mathbf{4.67 \times 10^{-2}}$
	All	25.02 ± 15.93	$\mathbf{14.35 \pm 8.82}$	$\mathbf{2.11 \times 10^{-3}}$
Manipulation Time (s)	EP_1	$\mathbf{8.47 \pm 3.84}$	11.75 ± 4.76	0.13
	EP_2	$\mathbf{8.53 \pm 5.31}$	12.37 ± 12.28	0.40
	EP_3	15.12 ± 16.80	$\mathbf{10.16 \pm 5.15}$	0.41
	All	$\mathbf{10.71} \pm 10.50$	$11.42 \pm \mathbf{7.90}$	0.80

significantly affect user performance for experienced users ($p_{AR} = 8.16 \times 10^{-2}$).

The results of RMS distance d_{RMS} are shown in Fig. 5.17c and Tab. 5.7. The mean d_{RMS} is reduced by 42.66%, from $25.02 \pm 15.93\,mm$ in II_{NA} to $14.35 \pm 8.82\,mm$ in II_{AR}. This reduction is achieved for all tested endoscope poses EP_1, EP_2 and EP_3, and is significant for EP_2 ($p = 3.88 \times 10^{-2}$) and EP_3 ($p = 4.67 \times 10^{-2}$). The two-way ANOVA shows that, with *ARssist*, the reduction of d_{RMS} is quite significant

($p_{AR} = 2.11 \times 10^{-3}$). The statistics reveal that, in a situation with larger misorientation problem (the hand-eye coordination is hindered to a larger extent in EP_2 and EP_3), *ARssist* is able to help the experienced users to keep an insertion trajectory with small deviation. However, in a normal situation where the hand-eye coordination is not an issue, the experienced users perform better with the standard setup.

Table 5.8: Subjective rating results of instrument insertion for the user study at Intuitive Surgical Inc. with experienced users ($N = 10$), corresponding to Fig. 5.17e. Data is represented as mean ± standard deviation. Better results are highlighted in bold font. Higher rating for "Fatigue" corresponds to less fatigue.

Metric	Non-AR	*ARssist*	p-value
Outcome	4.00 ± 0.67	$\mathbf{4.80 \pm 0.42}$	$\mathbf{4.89 \times 10^{-3}}$
Speed	3.30 ± 1.06	$\mathbf{4.50 \pm 0.70}$	$\mathbf{8.04 \times 10^{-3}}$
Confidence	3.15 ± 0.88	$\mathbf{4.45 \pm 0.68}$	$\mathbf{1.73 \times 10^{-3}}$
Satisfaction	3.50 ± 0.85	$\mathbf{4.60 \pm 0.52}$	$\mathbf{2.57 \times 10^{-3}}$
Fatigue	3.70 ± 1.34	$\mathbf{3.90 \pm 0.88}$	0.70
Interest	3.30 ± 0.82	$\mathbf{4.40 \pm 0.52}$	$\mathbf{2.14 \times 10^{-3}}$
Coordination	3.20 ± 1.32	$\mathbf{4.20 \pm 0.79}$	5.41×10^{-2}

We extract the participants' subjective ratings for their experience with and without *ARssist* from the questionnaire, as shown in Fig. 5.17e and Tab. 5.8. Again we assume that the result for each metric follows the normal distribution, then we use paired t-test to determine whether the results for two experiment setups differ. The t-test shows that users significantly prefer II_{AR} in *outcome*, *speed*, *confidence*, *satisfaction*, and *interest*. In terms of *fatigue* and *hand-eye coordination*, the average ratings in II_{AR} are not significantly higher than in II_{NA}, but are still showing

average improvements. We can conclude from the subjective ratings that *ARssist* improves the perception, ergonomics, self-assess performance and satisfaction, even for experienced users, to a moderate extent.

5.9.1.2 Tool Manipulation

The result for tool manipulation time t_{Mani} is shown in Fig. 5.17d and Tab. 5.7. It is $11.42 \pm 7.90\,s$ in TM_{AR}, which is actually slightly longer than $10.71 \pm 10.50\,s$ in TM_{NA} (6.70%). The average time reduction is only achieved with EP_3. Unlike the inexperienced users who commonly have difficulty navigating under bad hand-eye coordination, the experienced users have been more or less trained with such operation.

Table 5.9: Subjective rating results of tool manipulation for the user study at Intuitive Surgical Inc. with experienced users ($N = 10$), corresponding to Fig. 5.17f. Data is represented as mean ± standard deviation. Better results are highlighted in bold font. Higher rating for "Fatigue" corresponds to less fatigue.

Metric	Non-AR	*ARssist*	p-value
Outcome	$4.10 \pm \mathbf{0.57}$	$\mathbf{4.50} \pm 0.97$	0.28
Speed	3.50 ± 1.35	$\mathbf{4.20 \pm 1.14}$	0.23
Confidence	$3.20 \pm \mathbf{0.63}$	$\mathbf{4.30} \pm 0.95$	$\mathbf{6.88 \times 10^{-3}}$
Satisfaction	$3.80 \pm \mathbf{0.79}$	$\mathbf{4.40} \pm 1.07$	0.17
Fatigue	$\mathbf{3.90 \pm 1.10}$	3.60 ± 1.26	0.58
Interest	$3.70 \pm \mathbf{0.67}$	$\mathbf{4.10} \pm 0.88$	0.27
Coordination	3.00 ± 1.33	$\mathbf{4.40 \pm 0.97}$	$\mathbf{1.50 \times 10^{-2}}$

The subjective results for tool manipulation are shown in Fig. 5.17f and Tab. 5.9.

The experienced users reported better confidence and hand-eye coordination with *ARssist*, and the improvement is significant, even though the temporal performance did not improve. The users reported that there was more *fatigue* with *ARssist*; this is partly due to the fact that, they were able to finish the task in a similar time, in which case, the improvement in hand-eye coordination did not outweight the tiredness caused by wearing HoloLens.

5.9.1.3 Other Feedback

Through the experiment with experienced users, we have received distinct feedback from them. Some of them are enthusiastic about the potential of *ARssist* and believe that it should be pushed to the operating room, whereas other users express negative feedback and believe the limitations of current AR platforms cannot justify the ergonomic improvements. The limitations include the weight of the headset. Sometimes, when the headset (HoloLens v1) is not properly worn, e.g. the weight is not balanced (excessive weight on the nose), it is quite frustrating to the users, especially experienced users, to operate with it. This could be alleviated by a careful training session, but there often is limited time available for training. This results in inconsistent performance of the user. If the user wears the headset improperly in a certain trial, it is likely that the performance and subjective feedback will be compromised.

5.10 Recent Development

Since the user studies at Johns Hopkins University and Intuitive Surgical Inc., we have been continuing the effort to improve the usability of the system. We have re-written the streaming pipeline of *ARssist* to take advantage of GPU-based encoding on the Ubuntu PC and GPU-based decoding on HoloLens. The streaming is able to run at $60Hz$ for stereo $720P$ laparoscopy. The tracking capability on HoloLens is also improved to $30Hz$ with the widest FOV of the camera. We also integrate *ARssist* with the newest da Vinci Xi robot, as shown in Fig. 5.18.

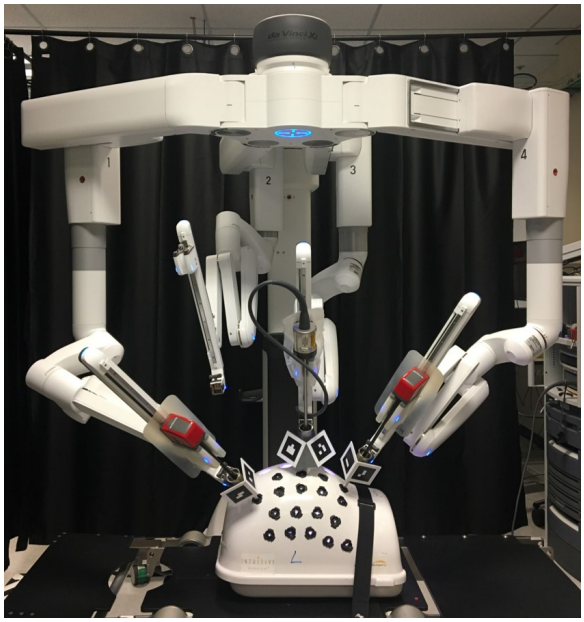


Figure 5.18: *ARssist* setup integrated with da Vinci Xi

We have made several demos since then, including at the Intuitive Research Symposium in January 2019, the LCSR Industry Day in March 2019, the International Conference on Robotics and Automation in May 2019 (Fig. 5.19a) and the Hamlyn

Symposium on Medical Robotics in June 2019 (Fig. 5.19b).

(a) At ICRA 2019, Montreal, Canada **(b)** At HSMR 2019, London, England

Figure 5.19: Demo of *ARssist* with da Vinci Xi

5.11 Open Source Contribution: dVRK-XR

During the development of *ARssist*, we implemented the kinematic streaming capability to support real-time visualization of a virtual robot on OST-HMDs (Sect. 5.3.3). In order to facilitate the integration of mixed reality for surgical robotics, we open sourced the package dVRK-XR, as a contribution to the dVRK community.

The system architecture of dVRK-XR is shown in Fig. 5.20. The dVRK employs a component-based software architecture [122], in which different modules can be dynamically loaded with a JSON-based configuration. In fact, this feature enables us to extend the dVRK software stack in a clean and reliable way. We implemented a new component: *sawSocketStreamer*, which can be connected to the existing dVRK program and send JSON-serialized messages at a fixed framerate over UDP.

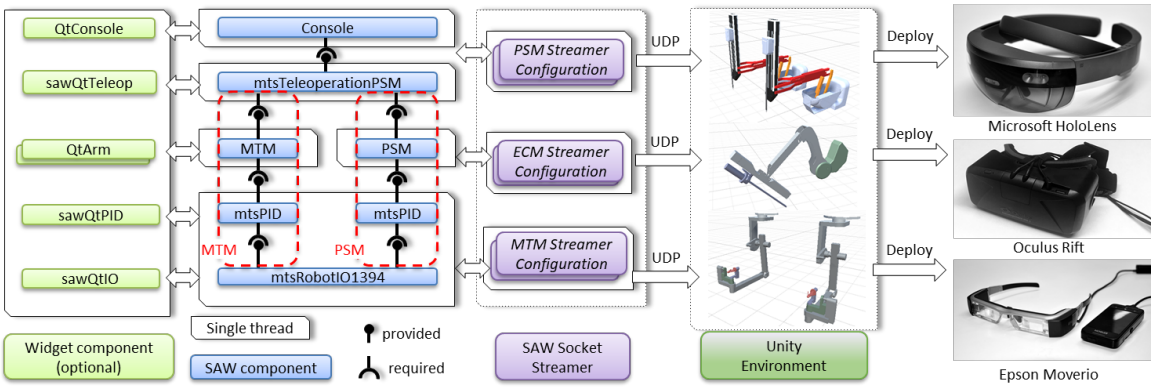


Figure 5.20: Overview of dVRK-XR open source package and integration with existing dVRK software stack

The mixed reality application is built with Unity, which can receive and deserialize the incoming messages containing the robot status information. We provide pre-fabricated PSM, ECM and MTM for dVRK-XR. With real-time messages containing joint state, the visualization of the manipulators can be synchronized with the real robot.

5.12 Conclusion

In this chapter, we developed and evaluated *ARssist* [207], an AR application to aid the first assistant in RAS. We chose two frequently occurring tasks, instrument insertion and tool manipulation, and conducted a series of user evaluations. In the pilot run, the experienced users did not show a significant difference with *ARssist*, but they agreed that it could be beneficial for the FA's performance, especially for inexperienced users. We refined the implementation and conducted a user study at

Johns Hopkins University with 20 inexperienced users, which showed that *ARssist* can significantly benefit inexperienced users by improving efficiency, navigation consistency, and safety for the instrument insertion task, and by enhancing hand-eye coordination for the tool manipulation task.

After gaining the insight from the previous evaluations, we conducted a user study at Intuitive Surgical Inc. with 10 relatively experienced users. The results demonstrated significant improvement in navigation safety and subjective preference in terms of hand-eye coordination. The users reported more fatigue with *ARssist*, largely due to the weight of the OST-HMD. For both tasks, the experienced users rated higher confidence level with *ARssist*. It demonstrated that the additional information, e.g. the robotic instrument and endoscopic field-of-vision, are useful in guiding the experienced user's decision and operation. Valuable feedbacks have been collected with our evaluation, that can further guide us to improve our system.

5.13 Closing Remarks

ARssist is our attempt to integrate OST-HMD-based AR for robotic surgery. Little prior work has been done in this domain. It has received positive feedback in our user studies and during exhibition. The improved hand-eye coordination is the most significant factor. The significantly improved confidence level indicated that the AR visualization of *ARssist* is able to provide useful guidance to even the experienced

users. The weight of the OST-HMD is still a frequently reported ergonomic drawback of *ARssist*. One limitation is the insufficient time of training with the AR system. For experienced users, they have been trained for operating instruments under different situations, however, there often is limited time for training to use *ARssist* before conducting the experiments. Hence, their proficiency of using *ARssist* is far from being at the same level as the traditional setup. It would be critical to conduct more studies after sufficiently training the users. In addition, much more effort is still required for *ARssist* to bring benefit to surgeries, both from technical and clinical aspects. The next chapter will present our technical enhancement to *ARssist*.

5.14 Published Work

Materials from this chapter appear in the following publications:

1. **Long Qian**, Anton Deguet, Peter Kazanzides, "dVRK-XR: Mixed Reality Extension for da Vinci Research Kit," *Hamlyn Symposium on Medical Robotics (HSMR)*, pp. 93-94. 2019.
2. **Long Qian**, Anton Deguet, Zerui Wang, Yun-hui Liu, Peter Kazanzides, "Augmented Reality Assisted Instrument Insertion and Tool Manipulation for the First Assistant in Robotic Surgery," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5173-5179. IEEE. 2019.
3. **Long Qian**, Anton Deguet, Peter Kazanzides, "ARssist: Augmented Reality on a Head-Mounted Display for the First Assistant in Robotic Surgery," *Healthcare Technology Letters (HTL)*, Volume 5, Issue 5, pp. 194-200. IET. 2018.

Chapter 6

ARAMIS: AR Assistance for Minimally-Invasive Surgery

In this chapter, we introduce *ARAMIS*, another OST-HMD-based AR application targeted at minimally invasive laparoscopic surgery. *ARAMIS* can provide real-time "x-ray see-through vision" of a patient's internal structure to the surgeon. It can also be applied for the first assistant in robotic-assisted surgery. In this case, *ARAMIS* differs from *ARssist* in that it provides a fully reconstructed 3D visualization instead of rendering laparoscopic video as a virtual monitor or as a frustum projection.

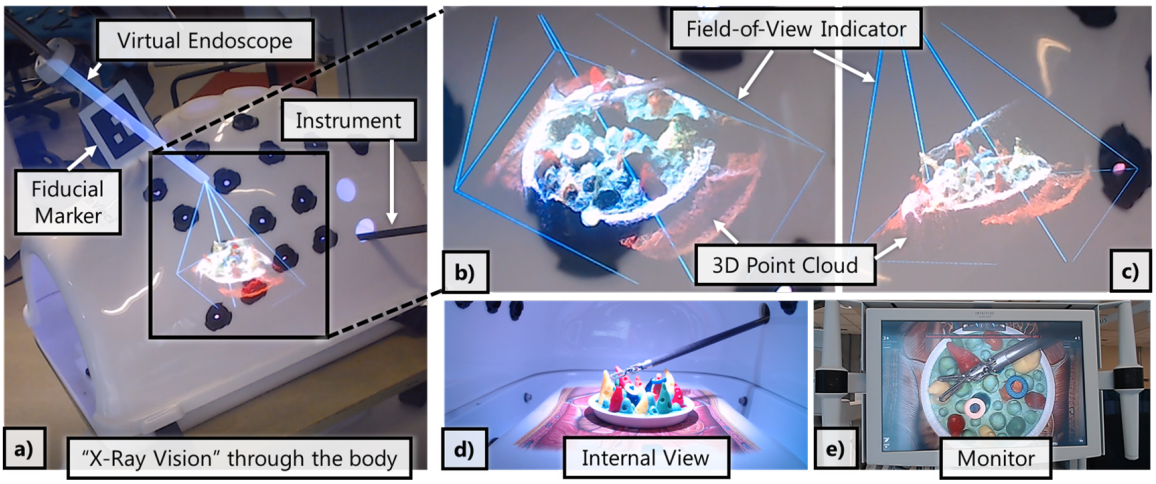


Figure 6.1: a) The "x-ray vision" provided by *ARAMIS*, captured using a camera behind HoloLens. b, c) Closer views of the 3D point cloud overlay. d) An alternative view inside the body phantom. e) Traditional monitor for laparoscopy.

6.1 Introduction

Laparoscopic surgery, also known as minimally invasive surgery (MIS) in the abdomen, has numerous advantages over the traditional open surgery, including smaller abdominal incision, reduced trauma, and preventing undue blood loss [44]. In laparoscopic surgery, a laparoscope or endoscope is inserted through a keyhole on the patient's abdominal wall. The real-time image captured by the endoscope is displayed on a monitor, and based on the visual guidance, the surgeon manipulates laparoscopic instruments, such as graspers and scissors, to perform the procedure. Although laparoscopic surgery has become the preferred approach for various procedures, it impairs the ergonomics and perception of the surgeon due to: i) the fulcrum effect, ii) the mislocation and misorientation of the endoscope display [286], and iii) poor depth visualization [29].

Robotic-assisted surgery and AR-assisted surgery provide two solutions to address the ergonomic issues of laparoscopic surgery. The discussion about the robotic-assisted approach is provided in Appendix A. Although robotic surgery has achieved great success, it requires extra training of the entire surgical team and has been considered more expensive [290]. AR is able to provide "x-ray vision" to the surgeon, where the real-time imaging of the anatomy is displayed at the correct position and depth, superimposed on the surgeon's normal vision, regardless of the viewing perspective of the surgeon.

Fuchs et al. first proposed to use a VST-HMD to achieve "x-ray vision" in laparoscopic surgery [70]. The surgery scene is reconstructed by structured-light methods, which introduces a considerable amount of latency. Moreover, using VST-HMD is not fail-safe in contrast to OST-HMD [218], where the surgeon can always look at the normal laparoscopic monitor. The resolution of the system was also limited by the capability of hardware in the 1990s. Since then, the literature has focused more on in-situ visualization of pre-operative models with different types of AR media, e.g., integral videography [142], instead of AR visualization of the laparoscopy itself. In the previous chapter, we proposed *ARssist* to visualize the instruments and endoscopic video inside the patient body in a robotic surgery, but rendering the laparoscopy as a 2D plane does not fully restore the depth perception [207].

There are many challenges towards achieving "x-ray vision" in laparoscopic surgery, for example:

- the surgery scene is highly deformable and dynamic
- the surgeon's motion and viewing perspective are unrestricted
- the rendering should be real time and high quality as visual feedback
- the augmented reality system needs to be fail-safe.

We propose, develop and evaluate *ARAMIS*, an AR system providing real-time "x-ray vision" in laparoscopic surgery based on an OST-HMD, as shown in Fig. 6.1, which addresses all the above technical challenges.

6.2 Contributions

The contribution of this chapter is:

1. We develop *ARAMIS*, an OST-HMD based AR application for the laparoscopic surgeon, enabling "see-through surgery". The efficacy of *ARAMIS* is evaluated in a simulated peg transfer procedure. *ARAMIS* provides improved hand-eye coordination through the in-situ low-latency 3D visualization via point cloud. The bandwidth-efficient representation of the point cloud and utilizing GPU computing on the HoloLens to decode the point cloud are the keys to the low latency of *ARAMIS*. Xiran Zhang assisted me in developing the CUDA-accelerated disparity calculation from stereo endoscopic images.

6.3 System Overview

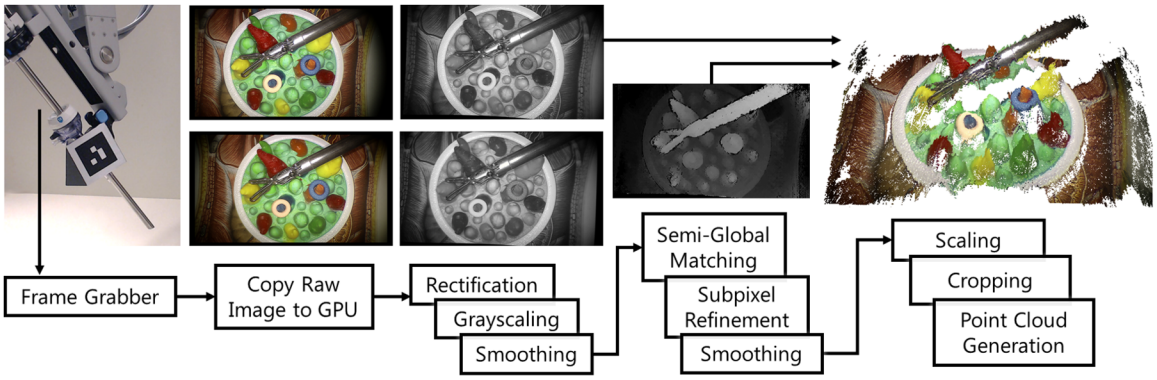


Figure 6.2: Image processing pipeline in *ARAMIS* to generate real-time point cloud

We used the da Vinci Si binocular endoscope (calibrated as a stereo camera) and Microsoft HoloLens for the implementation of *ARAMIS*. The endoscope is configured to output a stereo $720P$ image at $60\,Hz$, which is captured with a Blackmagic Duo 2 frame grabber. The images are immediately copied to an Nvidia Titan GPU where further processing takes place. An overview of the image processing pipeline is shown in Fig. 6.2. The method for point cloud representation, streaming and rendering in *ARAMIS* is shown in Fig. 6.3.

6.4 Methods

6.4.1 GPU-Accelerated Semi-Global Matching

Stereo matching is the technique to find the corresponding points in a stereo image pair (I_L, I_R). The image pair is first rectified using the stereo camera calibration

(I'_L, I'_R), so that the corresponding points of the left and right images are aligned horizontally. The rectification limits the corresponding point search region from $2D$ to $1D$ (horizontal). After that, the rectified image pair is smoothed using a Gaussian kernel, and converted to grayscale.

For each point on the rectified left image $I'_L(i, j)$, we compare the point on the rectified right image in the same row, with a maximum search range of 64 pixels $I'_R(i - c - d, j),\ d < 64$. The number c is a constant offset that we applied to adjust the minimum disparity value to search for. Census transform is used as a similarity metric when comparing two pixels. As an important concept of semi-global matching, the disparity value of nearby pixels should be similar, and a common approach to achieve this is to include a penalization term aggregating the cost along multiple paths towards the target pixel [103]. With a pre-computed table of similarity scores, the cost aggregation of each path direction can be performed asynchronously with GPU computation. A post-processing sub-pixel refinement is performed to interpolate the integer-valued disparity map D $(0 \sim 64)$ to increase precision. The resulting point cloud can be scaled down before being packed into a data buffer. The edges of the point cloud are discarded because the pixels at the border are noisy and unstable.

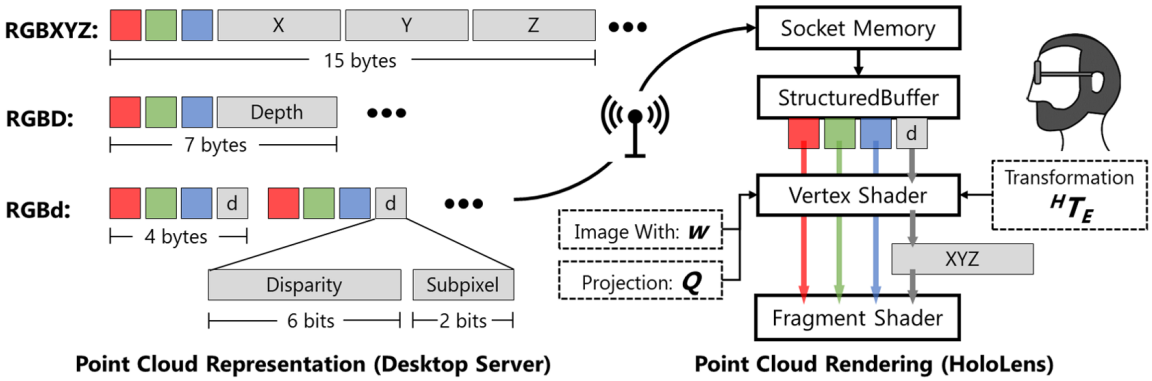


Figure 6.3: The point cloud representation, streaming and rendering in *ARAMIS*

6.4.2 Dense Point Cloud Representation, Streaming and Rendering

The $3D$ position of a point (i, j) in the disparity map D, with respect to the camera coordinate system, can be calculated with a projection matrix Q:

$$\begin{bmatrix} p_x \\ p_y \\ p_z \\ p_w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & 1/b & 0 \end{bmatrix} \begin{bmatrix} i \\ j \\ D(i,j) \\ 1 \end{bmatrix} \tag{6.1}$$

where (i, j) is the pixel coordinates of the target point, c_x, c_y, f are the left camera principle point and focal length, and b is the baseline distance of the stereo camera. The result is represented in homogeneous coordinates.

Conventionally, each point in the point cloud is stored in $RGBXYZ$ or $RGBD$ format. $RGBXYZ$ requires 15 bytes per pixel: a 3-byte color component and 3 floating point numbers for position. For $RGBD$, the depth value of each pixel is calculated with p_z/p_w of Eq. 6.1, requiring a total of 7 bytes per pixel. In contrast, we store the point cloud in a flattened $RGBd$ array, where d refers to disparity instead

of depth. We use 1 byte for the disparity value, with 6 bits contributed by the semi-global matching algorithm and 2 bits from the sub-pixel refinement. $RGBd$ is a more compact representation that preserves the precision of the disparity value. Prior to the streaming of point cloud data, the image width w and the projection matrix Q are sent to the HoloLens in order to compute the position of each pixel from the disparity value.

Upon receiving a complete buffer containing point cloud data on HoloLens, the data is uploaded to a pre-allocated StructuredBuffer on the GPU. We implement a custom shader to render the received data packet. We choose point as a render primitive. For each $RGBd$ datum, Eq. 6.1 is performed in the vertex shader with parameters w, Q, disparity value d, and the current transformation between the OST-HMD and the endoscope tip ${}^{H}T_{E}$. The RGB component is parsed in the fragment shading stage. The rendering pipeline is shown in Fig. 6.3.

6.4.3 Localizing the Endoscope Tip

We attach a fiducial marker to the endoscope, outside of the cannula. We denote the coordinate system of the OST-HMD as $\{H\}$, fiducial marker as $\{M\}$, endoscope tip as $\{E\}$, and the world as $\{W\}$, as shown in Fig. 6.4.

In order to render the point cloud with the correct pose while allowing the surgeon to freely walk around, the transformation ${}^{H}T_{E}$ (between endoscope tip and OST-HMD) is required at every frame of rendering. ${}^{H}T_{E}$ can be determined in two ways

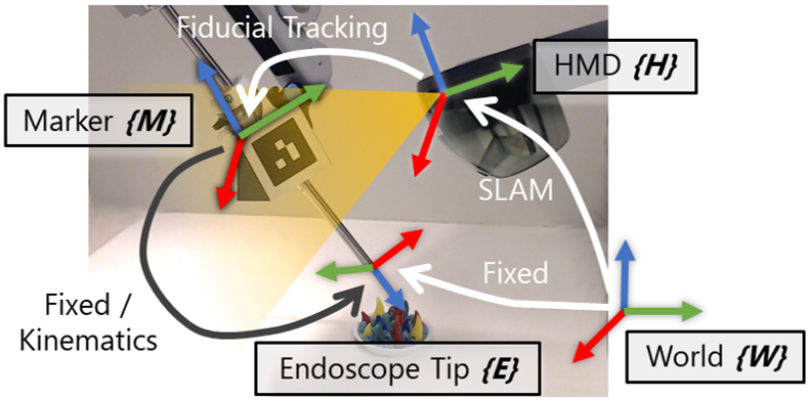


Figure 6.4: The transformation between each component in *ARAMIS*, ${}^{H}T_{E}$ is necessary for real-time point cloud rendering

(via SLAM or via fiducial tracking):

$$ {}^{H}T_{E} = {}^{H}T_{W}\,{}^{W}T_{E} \quad (2) \quad or \quad {}^{H}T_{E} = {}^{H}T_{M}\,{}^{M}T_{E} \quad (3) \tag{6.2} $$

where ${}^{M}T_{E}$ is known either by a pivot calibration or by kinematics if the endoscope is in a mechanical linkage. We apply the priority-based sensor fusion technique same as Sect. 5.3.2 and display calibration same as Sect. 2.6.

6.5 System Evaluation

6.5.1 Overlay Accuracy

We position a smartphone, showing a white crosshair, under the binocular endoscope. With *ARAMIS*, a point cloud of the crosshair is displayed on the OST-HMD as well. We capture the see-through view using a camera (Fig. 6.5a). The centers of the crosshairs are represented as C_1 and C_2 in the image space. We back-project

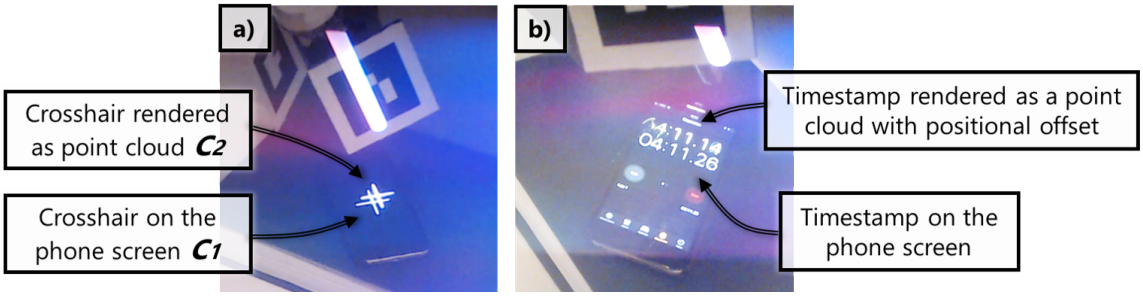


Figure 6.5: The setup for evaluating overlay accuracy (a) and end-to-end latency (b). A positional offset is applied in (b) so that both timestamps can be seen clearly.

the two crosshair centers to two rays ($\overrightarrow{r_1}$ and $\overrightarrow{r_2}$) intersecting with the camera center, and then calculate the visual angular error as the angle between the two rays:

$$\theta = arccos((\overrightarrow{r_1} \cdot \overrightarrow{r_2})/(\|\overrightarrow{r_1}\| \cdot \|\overrightarrow{r_2}\|)) \tag{6.3}$$

. We captured 32 overlay images from different poses ($^{H}T_{E}$), and the visual angular error is calculated to be 0.53°, with a standard deviation of 0.15°. At a distance of $0.5\,m$, which is a typical working range of a laparoscopic surgeon, the visual error will be $4.6\,mm$. It is noticeable that the error reported is the absolute error between the point cloud and the real object. Relative error between multiple objects, for example, between the laparoscopic instrument and the surgery scene in the point cloud, is much smaller. The overlay accuracy result is similar to the reprojection error reported by the display calibration method of HoloLens [206].

6.5.2 End-to-End Latency

A timestamp is displayed on the smartphone screen and therefore visualized on the point cloud on the HoloLens. The difference between the two timestamps reveals

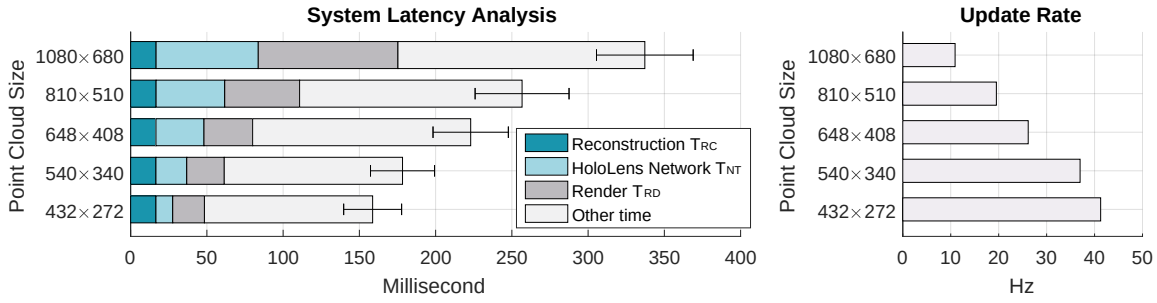


Figure 6.6: The end-to-end latency T and its decomposition, and update rate R with respect to different point cloud sizes (number of points) in *ARAMIS*

the end-to-end latency T. We measure the latency for multiple levels of point cloud scaling. As shown in Fig. 6.6, the end-to-end latency is $337.2 \pm 31.7\,ms$ for 1080×680 points, $256.7 \pm 30.8\,ms$ for 810×510 points, $223.0 \pm 24.7\,ms$ for 648×408 points, $178.3 \pm 21.0\,ms$ for 540×340 points, and $158.7 \pm 19.0\,ms$ for 432×272 points. With the largest point cloud, the update rate is $10.91\,Hz$. The update rate increases when the point cloud is scaled down. *ARAMIS* is able to achieve a $26.16\,Hz$ update rate when the point count is 648×408, $36.98\,Hz$ with 540×340 points, and $41.27\,Hz$ with 432×272 points.

We further analyze the time that is spent on point cloud reconstruction T_{RC} and client-side networking. The starting point for client-side networking T_{NT} is the time when HoloLens starts receiving the point cloud, instead of when the server starts sending, because the latter would require precise synchronization of clocks on both systems. T_{NT}, and rendering T_{RD}, are shown in Fig. 6.6. The time for reconstruction on the desktop server is almost constant for different numbers of points in the point cloud because the scaling is a post-processing procedure. On the contrary, the time

Figure 6.7: a) The user study setup with sample visualization of *ARAMIS*; b) the peg transfer task

for streaming the point cloud and rendering increases significantly with the size of the point cloud. "Other time" is calculated by $T - T_{RC} - T_{NT} - T_{RD}$, which includes the time spent by the endoscopic imaging (exposure), the frame grabber and system overhead. For reference, the end-to-end video latency of da Vinci S was measured to be $56.54 \pm 4.67\,ms$ [18].

6.6 User Evaluation

6.6.1 User Evaluation Setup

We chose peg transfer as the evaluation task, which is a typical task for laparoscopic skills evaluation. We aim to compare the user's performance of peg transfer in **normal laparoscopic setup** (traditional monitor), with the guidance of ***ARssist*** and with the guidance of ***ARAMIS***. In the experiment condition with *ARssist*, the

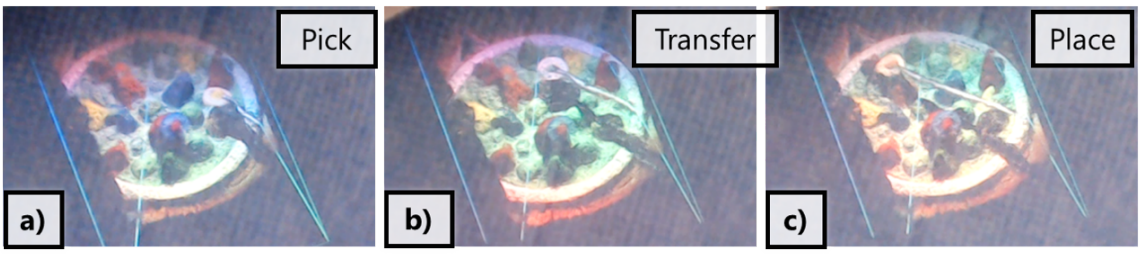


Figure 6.8: Guidance with *ARAMIS* during peg transfer on a deformable phantom

laparoscopic video is visualized as frustum projection (as shown in Fig. 5.3f), so as to control the difference between *ARAMIS*. The other two visualization methods in *ARssist* are disabled. Fig. 6.7b demonstrated the procedure of the peg transfer task. A deformable seaspike plate is placed inside the body phantom, with rubber rings placed on top of the spikes. The user manipulates a laparoscopic gripper to transfer the two pegs from one spike to the other. The body phantom has multiple incision ports on the side. For each setup, the user does the peg transfer task from three different incision ports, which represent different situations of hand-eye coordination.

6.6.2 User Evaluation Procedure

We recruited 26 users, including 3 experienced users. The study is approved by the JHU Homewood IRB. The detailed experiment procedure for each user is:

1. The user fills in a pre-experiment survey.
2. The user is trained to use a laparoscopic gripper to perform peg transfer with the traditional setup. He/she tries to perform the task from different ports.
3. The user is trained to use AR applications (*ARssist* and *ARAMIS*).

4. The order of experiments (lap. vs. *ARssist* vs. *ARAMIS*) is assigned to the user.
5. The user performs peg transfer twice (two rubber rings, as shown in Fig. 6.8b) from three different ports, under the guidance of system No.1.
6. The user fills out the post-experiment survey about the experience of using system No.1.
7. Steps 5 and 6 are repeated both for system No.2 and system No.3.

The post-experiment questionnaire collects the task load index (NASA-TLX) of each experiment, and the user's subjective rating of *outcome*, *speed*, *confidence*, *satisfaction*, *fatigue*, *interest* and *hand-eye coordination* with the specific setup. The ratings range from $0 \sim 5$. The post-experiment survey is the same as the study in Sect. 5.6.4.

Therefore, there are 234 trials in total (26 subjects $\times$ 3 ports $\times$ 3 setups). The laparoscopic videos were recorded. The time to complete the task, number of failure cases, the task load index and the subjective ratings were processed for analysis and discussion in the next session.

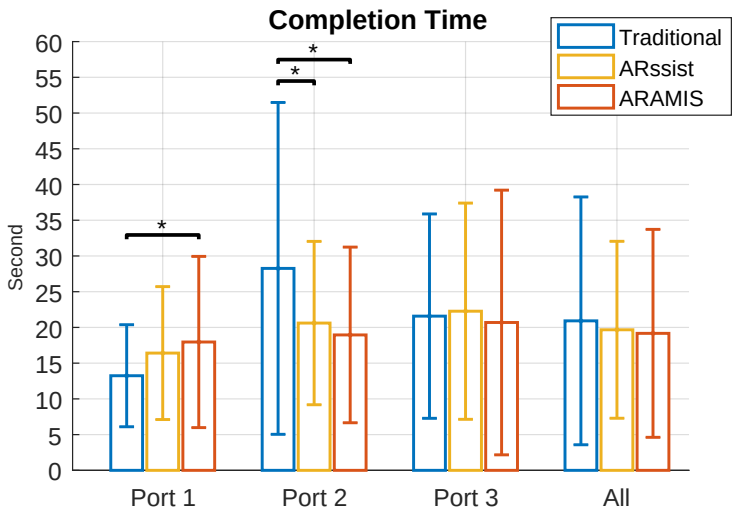


Figure 6.9: The completion time of peg transfer for three setups, assessed from three different ports. ($N = 26$)

6.7 Results and Discussion

6.7.1 Data of All Users

There were in total 11 failed cases for the laparoscopic setup, 8 for *ARssist* and 8 for *ARAMIS*. Fig. 6.9 and Tab. 6.1 demonstrate the average time to complete the peg transfer task under three different setups. In average, the users spent $20.92 \pm 17.34s$ with the normal laparoscopic setup, $19.67 \pm 12.38s$ with the guidance of *ARssist*, and $19.17 \pm 14.55s$ with the guidance of *ARAMIS*. The AR systems achieved in average 5.98% and 8.37% improvement over the traditional laparoscopic setup.

We first use Shapiro-Wilk test to determine whether the results of completion time follow a normal distribution. The null hypothesis is rejected for all data groups (per port per setup). Then for each port (EP_1, EP_2, and EP_3), we apply paired t-test

Table 6.1: The completion time of peg transfer for three setups, assessed from three different ports, corresponding to Fig. 6.9 ($N = 26$). Data of completion time are represented as mean ± standard deviation. The p-values are the results of paired t-test with traditional condition. Better values are highlighted in bold font.

Port	Traditional	*ARssist*	*ARAMIS*	p-value *ARssist*	p-value *ARAMIS*
EP_1	$\mathbf{13.24 \pm 7.14}$	16.41 ± 9.30	17.96 ± 11.98	5.95×10^{-2}	$\mathbf{1.91 \times 10^{-2}}$
EP_2	28.26 ± 23.22	$20.61 \pm \mathbf{11.43}$	$\mathbf{18.95} \pm 12.29$	$\mathbf{4.53 \times 10^{-2}}$	$\mathbf{1.70 \times 10^{-2}}$
EP_3	$21.58 \pm \mathbf{14.30}$	22.27 ± 15.13	$\mathbf{20.69} \pm 18.52$	0.82	0.80
All	20.92 ± 17.34	$19.67 \pm \mathbf{12.38}$	$\mathbf{19.17} \pm 14.55$	0.48	0.35

to determine whether the mean value of the completion time under each experiment condition differs. The null hypothesis is rejected when comparing *ARAMIS* and traditional setup for EP_1 and EP_2, and when comparing *ARssist* with traditional setup for EP_2. EP_1 is an incision port that does not create a significant hand-eye coordination challenge to the user. Therefore we actually observe that the users are more successful in the traditional setup than with *ARAMIS*. EP_2 is a port that does generally create hand-eye coordination issues in our experiment, where we can observe higher efficiency being achieved with the assistance of AR systems.

We further conduct an N-way ANOVA test to examine whether there is significant difference between the completion time of the peg transfer task under different experiment conditions. There are two categorical parameters here: the port (EP_1, EP_2, and EP_3) and the experiment condition (traditional, *ARssist* and *ARAMIS*). The N-way ANOVA does not reveal significant difference between experiment conditions

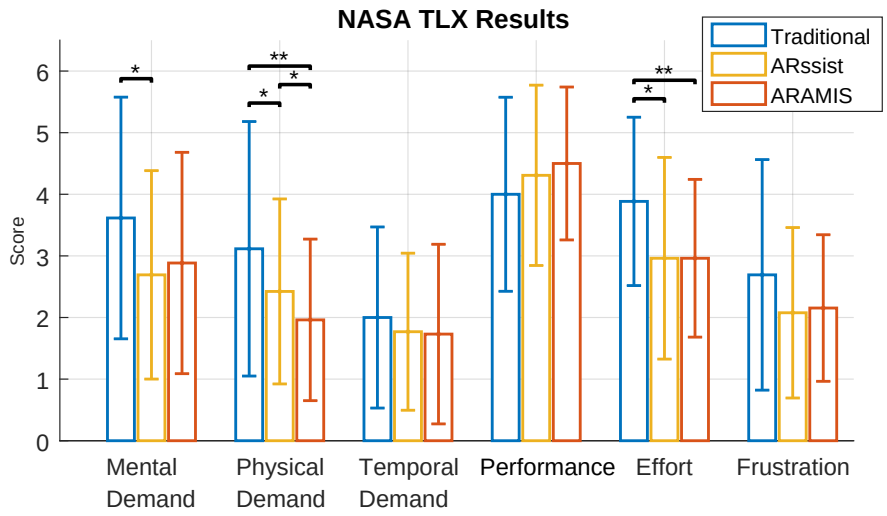


Figure 6.10: NASA Task Load Index results for three setups ($N = 26$)

($p = 0.36$). The ANOVAN test, however, found that the access port significantly alters the completion time ($p = 1.50 \times 10^{-3}$). It is expected because the hand-eye coordination of EP_1 is much more intuitive than EP_2 and EP_3.

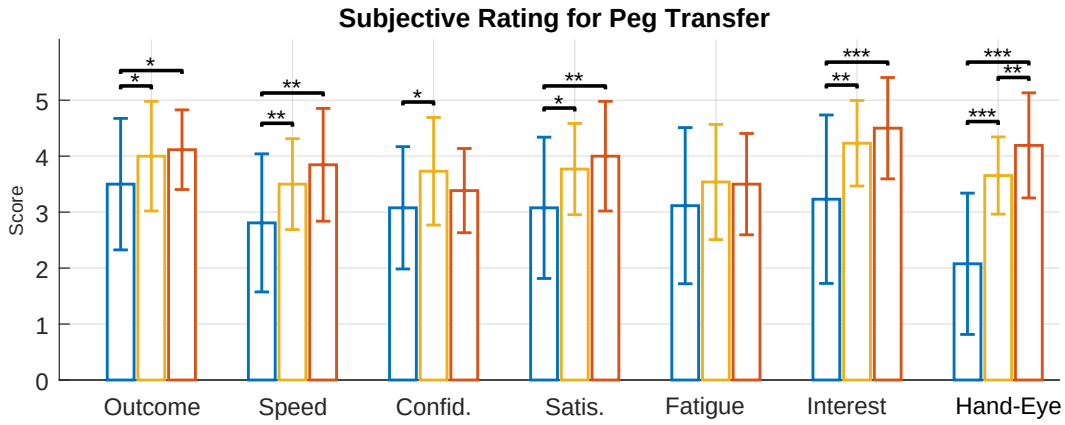


Figure 6.11: Subjective questionnaire results for three setups (N=26)

Fig. 6.10 and Tab. 6.2 show the task load index of the different surgical guidance methods. We can observe a general improvement by AR systems. We assume that the task load index results for each metric follow normal distribution, and then apply

Table 6.2: NASA Task Load Index results for three setups corresponding to Fig. 6.10. Data of task load are presented as mean ± std. The p-values are determined using paired t-test. The better number for each experiment setup is highlighted using bold font. (N=26)

Metric	Traditional	*ARssist*	*ARAMIS*	p-value *ARssist*	p-value *ARAMIS*
Mental	3.62 ± 1.96	$\mathbf{2.69 \pm 1.69}$	2.88 ± 1.80	7.52×10^{-2}	0.17
Physical	3.12 ± 2.07	2.42 ± 1.50	$\mathbf{1.96 \pm 1.31}$	0.17	$\mathbf{1.99 \times 10^{-2}}$
Temporal	2.00 ± 1.47	$1.77 \pm \mathbf{1.27}$	$\mathbf{1.73} \pm 1.46$	0.55	0.51
Perform.	4.00 ± 1.57	4.31 ± 1.46	$\mathbf{4.50 \pm 1.24}$	0.47	0.21
Effort	3.88 ± 1.37	$\mathbf{2.96} \pm 1.64$	$\mathbf{2.96 \pm 1.28}$	$\mathbf{3.19 \times 10^{-2}}$	$\mathbf{1.52 \times 10^{-2}}$
Frustra.	2.69 ± 1.87	$\mathbf{2.08} \pm 1.38$	$2.15 \pm \mathbf{1.19}$	0.18	0.22

paired t-test to compare different experiment conditions for each metric. The p-values are shown in Tab. 6.2 and are visualized as the horizontal star bars in Fig. 6.10. For physical demand and effort level, AR systems are both significantly better than traditional setup. In average, the frustration level is decreased with AR systems, but not significantly. The temporal demand and performance are similar for different setups, which is consistent with the reults of completion time in Fig. 6.9.

Fig. 6.11 and Tab. 6.3 show the subjective ratings collected from the user. The analysis method for the subjective ratings is similar to the task load index. We assume the normality distribution of the data, and apply t-test to compare different experiment conditions for each metric. The difference for hand-eye coordination is the most significant. The traditional setup has a rating of 2.08, while *ARssist* is 3.65 and *ARAMIS* is 4.19. *ARssist* is significantly better than the traditional setup ($p = 9.43\times$

Table 6.3: Subjective questionnaire results for three setups corresponding to Fig. 6.11 (N=26). Data are represented as mean ± standard deviation. Better results are highlighted in bold font. Higher rating for "Fatigue" corresponds to less fatigue.

Metric	Traditional	*ARssist*	*ARAMIS*	p-value *ARssist*	p-value *ARAMIS*
Outcome	3.50 ± 1.17	4.00 ± 0.98	$\mathbf{4.12 \pm 0.71}$	0.10	$\mathbf{2.66 \times 10^{-2}}$
Speed	2.81 ± 1.23	$3.50 \pm \mathbf{0.81}$	$\mathbf{3.85} \pm 1.01$	$\mathbf{2.07 \times 10^{-2}}$	$\mathbf{1.66 \times 10^{-3}}$
Confidence	3.08 ± 1.09	$\mathbf{3.73} \pm 0.96$	$3.38 \pm \mathbf{0.75}$	$\mathbf{2.62 \times 10^{-2}}$	0.24
Satisfaction	3.08 ± 1.26	$3.77 \pm \mathbf{0.82}$	$\mathbf{4.00} \pm 0.98$	$\mathbf{2.28 \times 10^{-2}}$	$\mathbf{4.89 \times 10^{-3}}$
Fatigue	3.12 ± 1.40	$\mathbf{3.54} \pm 1.03$	$3.50 \pm \mathbf{0.91}$	0.22	0.24
Interest	3.23 ± 1.50	$4.23 \pm \mathbf{0.76}$	$\mathbf{4.50} \pm 0.91$	$\mathbf{3.97 \times 10^{-3}}$	$\mathbf{5.63 \times 10^{-4}}$
Coordination	2.08 ± 1.26	$3.65 \pm \mathbf{0.69}$	$\mathbf{4.19} \pm 0.94$	$\mathbf{9.43 \times 10^{-7}}$	$\mathbf{1.01 \times 10^{-9}}$

10^{-7}), and *ARAMIS* is significantly better than the traditional setup ($p = 1.01 \times 10^{-9}$) and *ARssist* ($p = 2.24 \times 10^{-2}$). The result is consistent with our hypothesis that a full 3D representation of the surgical site yields the best perception (depth and coordination) for the user. Interestingly, the fatigue level is similar between the different setups. Although the weight of the OST-HMD is causing some fatigue, the perceptual benefit of AR systems is able to offset the disadvantage to a certain extent.

6.7.2 Data for Experienced Users

There are three experienced users recruited for the user study. Fig. 6.12 demonstrates the average time for experienced users to complete the peg transfer task under three different setups. In average, the experienced users spent $13.14 \pm 7.14s$

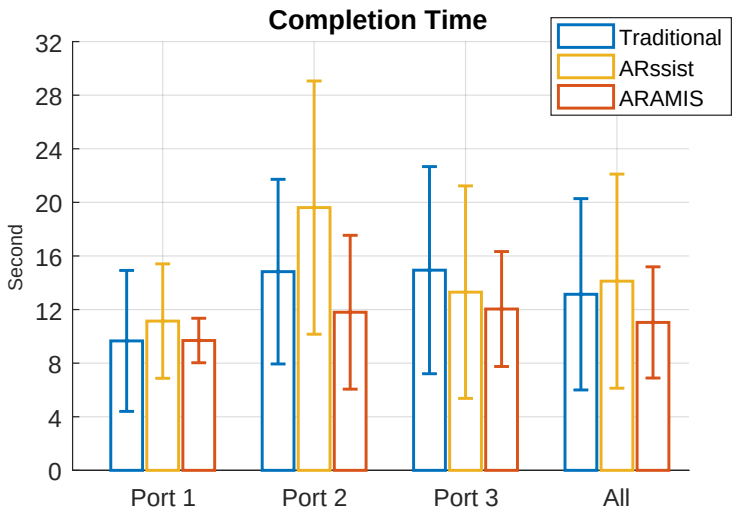


Figure 6.12: The completion time for experienced users for three setups, assessed from three different ports ($N = 3$).

with the normal laparoscopic setup, $14.12 \pm 799s$ with the guidance of $ARssist$, and $11.04 \pm 4.15s$ with the guidance of $ARAMIS$. The average completion time is decreased for $ARAMIS$, but due to the limited sample size, the differences are not significant for all the comparisons.

It was reported that the size of the point cloud visualization is limited with $ARAMIS$. In fact, the traditional monitor demonstrates a magnified view of the endoscopy, but since $ARAMIS$ visualizes the surgical site at its actual physical scale, it is not as clear as the monitor view. One solution to alleviate the "physical scale" problem is to scale up the point cloud, however, the scale factor in this case needs to be carefully selected, to avoid the loss of perceptual benefit of the see-through illusion. Another potential solution to alleviate this "physical scale" problem is to integrate a virtual monitor with $ARAMIS$. The virtual monitor displays the stereo laparoscopic

video with magnification, to allow the observation of details. At the same time, when the users feel difficulty with localization and hand-eye coordination, they can look at the point cloud visualization in *ARAMIS*.

6.8 Limitations and Future Work

From the technical point of view, *ARAMIS* is an enhanced version of *ARssist* where the stereo laparoscopic video is 3D real-time reconstructed into a point cloud, and the visualization on the OST-HMD is therefore fully 3D. However, visualizing the surgical site as a set of points causes some confusion to the user. Some holes exist when the surface is not well reconstructed, and due to the sparsity of points. One important piece of future work is to improve the quality of the reconstructed point cloud to enable finer details and to eliminate reconstruction artifacts, with techniques such as convex optimization [34], super-pixel segmentation [192] and deep-learning-based refinement [125].

Our implementation of *ARAMIS* relies on a binocular endoscope, but could also be integrated with a monocular endoscope (which is still more popular than binocular ones), using structured light or deep-learning-based reconstruction methods [149].

In order to prove the clinical benefit, more clinical tasks need to be evaluated with *ARAMIS*. Our future work includes a larger user study with both novice and experienced users, including surgeons, in a more realistic ex-vivo experimental setup.

6.9 Conclusion

In this chapter, we presented *ARAMIS*, which stands for augmented reality assistance for minimally-invasive surgery. It is another OST-HMD-based AR application, aiming to improve the ergonomics of the surgeon during laparoscopic surgery. *ARAMIS* relies on real-time 3D reconstruction from the binocular endoscope, real-time streaming and visualization of the point cloud data on an OST-HMD. Through our preliminary user study, we are able to prove that both experienced and inexperienced users are able to complete laparoscopic tasks with *ARAMIS*. Subjective results suggested that the hand-eye coordination is improved with the "see-through vision" enabled by *ARAMIS*.

6.10 Closing Remarks

ARAMIS is based on HoloLens 1st gen, which at the time of implementation, is the most advanced and popular research platform for optical see-through augmented reality. We have devoted much effort into optimizing *ARAMIS* to balance the user experience, which usually demands high framerate and low latency, and the computational burden. We are able to put together a reasonable application for evaluation, as a proof-of-concept. As can be observed from the evaluation results, the benefits are not very significant. However, in the future, I think many of the engineering constraints will be eliminated by more advanced hardware platforms. At that time,

I believe the actual benefit brought by AR technologies will be more obvious and convincing.

6.11 Published Work

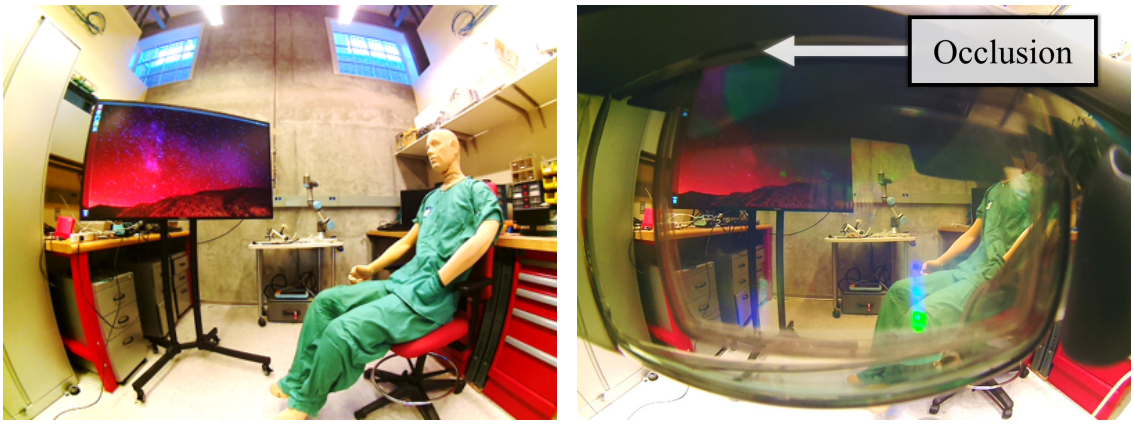
Materials from this chapter appear in the following publication:

1. **Long Qian**, Xiran Zhang, Anton Deguet, Peter Kazanzides, "ARAMIS: Augmented Reality Assistance for Minimally Invasive Surgery Using a Head-Mounted Display," *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pp. 74-82. Springer. 2019

Chapter 7

Restoring the Awareness Caused by OST-HMD Occlusion

This chapter presents technical contributions related to improving the safety of an OST-HMD-based AR application, more specifically, tackling the occlusion at the periphery caused by the OST-HMD. In surgical applications, which is the main focus of this thesis, OST-HMD occlusion can affect the situation awareness of the surgeon, e.g. blocking the view to an assistant. In this chapter, we first describe the occlusion issue with OST-HMDs, the importance of human peripheral vision, and the potential safety concerns for OST-HMD-based AR application. We then present our solution which includes both a hardware prototype and software algorithms to correctly restore the lost awareness, followed by experiments and evaluations.



(a) Usual view without occlusion **(b)** See-through view with HoloLens

Figure 7.1: OST-HMDs partially occlude the user's field of vision with the hardware structure. The occlusion causes the loss of awareness of the environment.

7.1 Introduction

OST-HMDs can be applied in critical situations like navigation, manufacturing and surgeries. However, while the form-factor of an OST-HMD occupies less of the user's visual field than in the past, it can still result in critical oversights. Fig. 7.1 demonstrates the see-through view with Microsoft HoloLens. OST-HMDs do not directly intercept the user's vision as VST-HMDs do, but they still introduce additional interference, e.g., distortion [113] and occlusion [291]. The distortion is caused by the structure of the optical system, as the optical elements in front of the user's eyes unavoidably cause refraction of light which can be estimated by an offline calibration procedure [113, 128]. However, this calibration cannot account for the portion of the user's peripheral view that is occluded by the OST-HMD [291] (Fig. 7.1b).

The occluded area is located in the peripheral vision, which is critical for safe and

efficient mobility [150]. The occlusion caused by the frame of the OST-HMD is thus a significant security risk, e.g., a pedestrian may not see a car coming from the side or a worker may miss a moving robot arm because it is occluded by the OST-HMD frame. With increasing AR applications built on OST-HMD platforms, it is important to face the issue of the incomplete awareness and alleviate the potential danger.

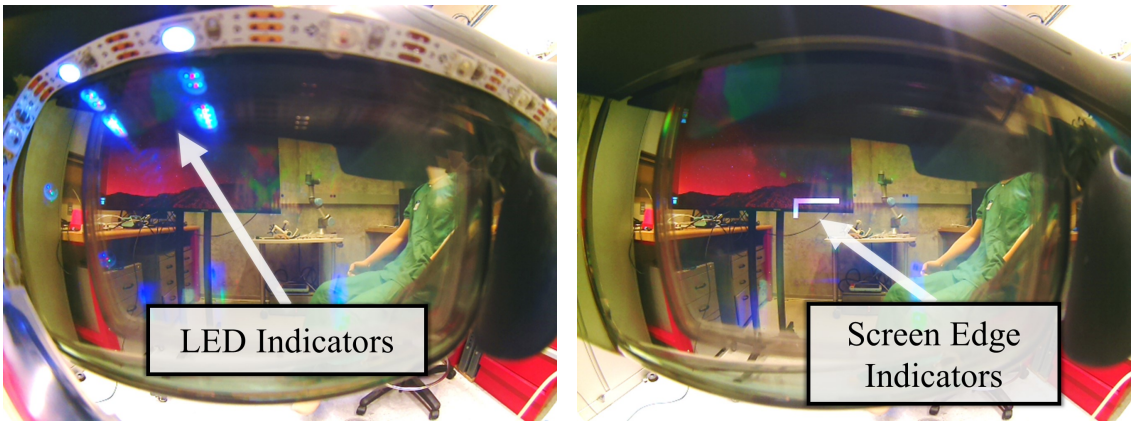
An ideal solution would be to use contact-lens type displays, e.g. Mojo Lens [168], however it is not clear when they will reach the consumer market. A common approach for VST-HMDs is to capture the invisible area with a camera and then map it onto the display [188, 9]. But this solution is not viable for OST-HMDs. We propose to use a wide-angle front-facing camera to capture the environment and then use alternative indicators (Fig. 7.2) to provide visual indications to the user about potential danger in the environment.

In the operating room, the awareness of the environment of the surgeon is critical. For example, the surgeon should be aware of the motion of the patient, or motion of the assistants. As another example, combat medics and paramedics who usually work in dangerous situations should be constantly aware about their surroundings.

7.2 Contributions

The contribution of this chapter is:

1. We develop a method to restore the lost peripheral awareness caused by the



(a) Usual view without occlusion **(b)** See-through view with HoloLens

Figure 7.2: We propose to use additional LEDs or the edge of the OST-HMD screen to restore the lost awareness.

occlusion of the hardware of OST-HMD. We model the occluded field-of-vision for a specific user and a specific OST-HMD, detailed in Sect. 7.4. We use LEDs or the screen edge of the OST-HMD as indicators of activity in the occluded field-of-vision. We calibrate the system so that specific indicators reflect the change in the environment in specific directions. Dr. Alexander Plopski proposed to use LED indicators for this method, and contributed to the writing of the manuscript.

7.3 Background and Literature Review

In this section, we first review the anatomy of the human eye and the user's field of vision. We then discuss previous studies that investigated effects of diminished field of vision, as well as previous work on extending the user's field of view in HMDs.

7.3.1 Human Visual Field

The field of vision is that portion of space in which objects are visible during gaze fixations [238]. The human visual field is usually measured by perimeters [181]. The visual field is dependent on the eye's health condition and the person's age.

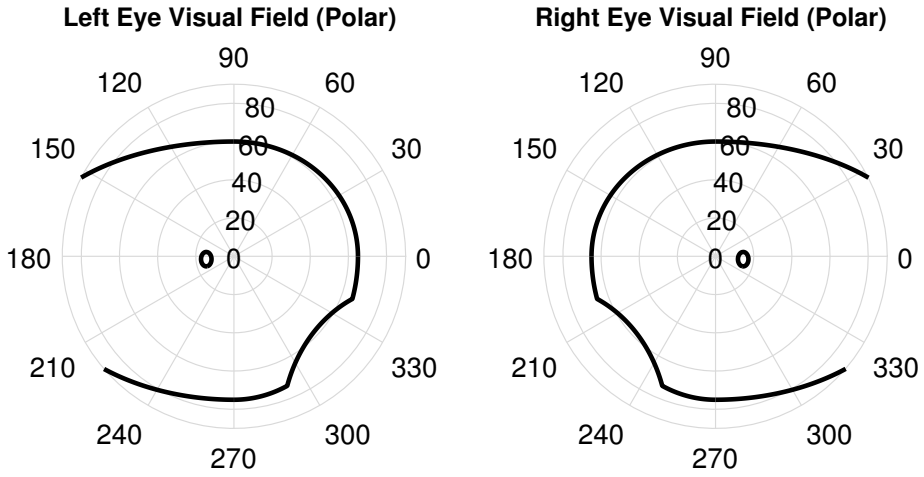


Figure 7.3: Sample of synthesized visual field for human left and right eye (V_{EL} and V_{ER}), represented in polar coordinate system.

A region in the visual field where no target can be seen is called absolute scotoma. Areas where some targets can be seen, but smaller ones are not visible are referred to as relative scotoma [291]. Fig. 7.3 shows a synthesized visual field in a polar coordinate system that captures the main features of the normal human visual field (typically 100° lateral, 60° medial, 60° upward and 75° downward span of visual field [238]; scotoma at the blind spot and nose; interpolation with elliptical curves). The blind spot is a portion of the human retina without any photoreceptors [82].

The human retina contains three types of photoreceptor cells that facilitate our

vision: rod, cone and non-image-forming photosensitive ganglion cells [48, 225]. Cones are responsible for the eye's color sensitivity. They are concentrated in the fovea centralis and there is only a small number of cones in the peripheral area of the eye. Rods are more sensitive to brightness than cones, but insensitive to colors. Their distribution is contrary to the rods, with only a few rods in the fovea centralis, and comparably many in the peripheral area. Therefore, while humans have very accurate vision of the focused area, we become less sensitive to details in the periphery. In fact, our ability to detect the color of objects in our periphery depends on the size of the stimuli [220].

7.3.2 Occlusion of Peripheral Vision and Danger

The hardware of an OST-HMD causes both absolute scotoma and relative scotoma to the user's visual field [291]. The occlusion is usually on the user's peripheral vision because the center part is designed for graphics overlays and direct (see-through) vision of the environment. Peripheral vision is critical for safe and efficient mobility [150]. Johnson et al. [117] studied the loss of visual field and its relationship to driving performance. They screened 10,000 drivers and found that drivers with binocular visual field loss had accident and conviction rates twice as high as those with normal visual fields. Szlyk et al. [250] found that driving performance of glaucoma patients correlates with peripheral visual field loss. Apart from keeping the person safe, peripheral vision also contributes to form vision [244] and scene gist recognition

by resolving lower spatial frequencies [137]. Researchers have also proposed using peripheral perception for conveying information [52, 130].

Therefore, when an AR application on an OST-HMD is used in a mobile scenario, or requires the user to pay attention to the surroundings, developers and designers must seriously consider the occlusion issue.

7.3.3 View Expansion with HMD

The loss of the visual field is a concern in VR and AR applications. Methods to expand the user's field-of-view (FOV) have been studied in the context of normal monitors [118], 3D monitors [30], mobile devices [257], heads-up displays [256], skier's helmet [182] and HMDs, in order to improve game experience or to facilitate localization of objects. We review literature that investigates view expansion with HMDs. The literature can be categorized by the type of HMD used and whether the expansion is applied to the real or the virtual environment. The taxonomy is shown in Tab. 7.1. VST-HMDs are more used for FOV expansion in the literature because of the easy access and full control of the user's view.

Table 7.1: Literature about view expansion on HMDs

Literature	VST-HMD	OST-HMD
FOV of virtual environment	[25], [49], [85] ,[86], [242], [247], [295]	[87], [295]
FOV of real environment	[9], [63], [164], [188], [263], [299]	[216], [263]

Expanding the FOV in the virtual environment is a kind of off-screen visualiza-

tion technique. The system is aware of the location of off-screen objects and provides hints to the user about their existence. Stoakley et al. [242] informed the user about objects not in the FOV through a WIM (World in Miniature) of the virtual environment on a VST-HMD in 1995. Gruenefeld et al. compared different off-screen visualization methods [85] (arrow [31], halo [21] and wedge [89]) for a VST-HMD, designed EyeSee360 [86] for VST-HMD and later integrated the same method to an OST-HMD [87]. In EyeSee360, a miniature world map is displayed to the user, and off-screen objects are plotted with variable color or size to indicate their distances. Boicca et al. [25] use Attention Funnel to guide the user's attention "down" the virtual funnel to the target location with the VST-HMD [25]. Sukan et al. [247] designed ParaFrustum to guide the user's viewing position and orientation of an object of interest on a VST-HMD. Xiao et al. [295] proposed the concept of sparse peripheral display, and integrated sparse LED lights into both a VST-HMD and an OST-HMD in order to facilitate object search and reduce motion sickness.

To expand the user's FOV in the real environment, Ardouin et al. [9] captured 360° FOV images and displayed them to the user on a VST-HMD. Fan et al. [63] proposed SpiderVision to analyze the back-view image and overlay it on the front-view image with a VST-HMD. Miyaki et al. proposed LiDARMAN, where the user sees a third-person view of himself/herself in a point cloud of the environment [164] on a VST-HMD. Orlosky et al. proposed Fisheye Vision to compress more peripheral FOV [188]. A dynamic view expansion method is also implemented to facilitate

visual search with a VST-HMD [299]. Vargas-Martin et al. proposed to visualize a minimized view of a wider FOV to aid people with restricted visual field due to retinitis pigmentosa and glaucoma [263]. They evaluated their method with VST-HMDs and OST-HMDs.

In addition to providing information about the environment, similar visualization technologies have also been used to display notifications [45, 151]. The visual language for peripheral display is investigated in [152]. Researchers have also investigated optical designs to expand the FOV of the display [35, 213, 212], while our solution is built upon existing commercial products.

The setups in [263, 216] appear closer to our work in that they all augment the real environment using an OST-HMD. In [263], an edge view is directly overlaid in the patient's central visual field, which may be intrusive for normal users. Renner et al. [216] simulated an OST-HMD using a VR headset and evaluated different off-screen visualization techniques for their searching efficiency. None of the above solutions considered the occlusion caused by the OST-HMDs, nor tried to alert the user about the potential danger caused by such occlusion.

7.4 Methods

The goal is to compensate for the loss of vision in the occluded areas of the OST-HMD. To do so, it is necessary to determine the invisible area for a user wearing an

OST-HMD, and to define a scheme to compensate for it. In this section, we introduce our approach. We record the area surrounding the user with a wide-angle camera CC that is attached to the OST-HMD. First, we explain the offline calibration process that determines what area of the image captured by CC is invisible to the user due to the occlusion by the OST-HMD. Second, we describe two methods that use two types of indicator to compensate the information loss in the occluded region of interest ($OROI$). We conclude with an explanation of how we visualize the information in an $OROI$ using the indicator.

We denote some extensively used objects for this chapter: left eye (EL), right eye (ER), center camera (CC), camera simulating left eye (CL), camera simulating right eye (CR). We refer to properties of these objects as the visual field (V), occlusion in visual field (O), camera intrinsic matrix (K), camera distortion function ($D(\cdot)$), indicator (I) and its associated occluded region of interest ($OROI$). We refer to properties of a particular object by writing the said object as a subscript of the property. For example, the visual field of left eye is V_{EL}. When we project information from one object to another, we denote the object it is projected into as a superscript. For example, V_{EL}^{CC} refers to the visual field of the left eye projected onto the visual field of the center camera.

7.4.1 Determine the Occluded Visual Field

Before we can compensate the information occluded by the frame of the OST-HMD, we have to first determine the occluded area. To do so, we divide the question "where to restore the awareness" into three sub-problems:

$Q1$**:** Which part of V_{CC} can the user see normally?

$Q2$**:** Which part of V_{CC} is occluded by the OST-HMD?

$Q3$**:** Which part of V_{CC} should the system compensate for the user?

7.4.1.1 Human visual field projected on camera visual field

In order to answer $Q1$, we need to project the normal human visual field onto the camera's visual field V_{CC}. The human visual field is usually measured by a perimeter [181], and the results are presented in a polar coordinate system, as in Fig. 7.3. We first transform the polar representation to a Cartesian coordinate system and then project it to the visual field of the camera.

To simplify the calculations we assume that:

$A1$**:** The user's eyes and the camera CC are co-located.

$A2$**:** The user's fixation is steady and the viewing direction coincides with CC.

$A1$ is equivalent to assuming that all light rays come from infinity. We discuss the error introduced by this assumption in Sec. 7.6. Assumption $A2$ could be removed with the help of eye-tracking methods that detect the eye's rotation in real-time.

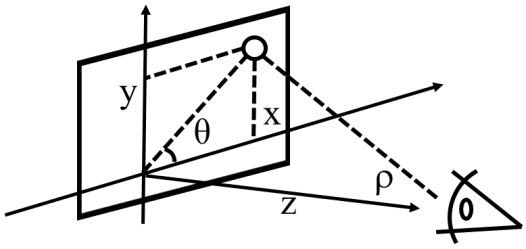


Figure 7.4: Transformation between polar coordinate system of human visual field and Cartesian coordinate system

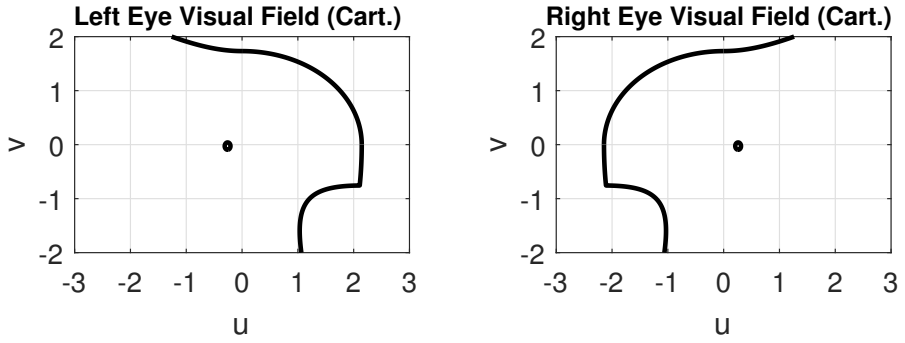


Figure 7.5: Sample visual field for human left and right eye (V_{EL} and V_{ER}) in Cartesian coordinate system. XY axes are (u, v).

Under $A1$ and $A2$, the coordinate systems of EL, ER, and CC are identical. It is thus sufficient to determine what pixel $p(i, j)$ in the image captured by CC corresponds to a given angle $p_{polar}(\rho, \theta)$, $0° \leq \rho < 90°$ of V_{EL} and V_{ER}. To determine p, we convert p_{polar} into the Cartesian coordinate system.

Assume a plane at a distance z in front of the eye. The light ray $\vec{L}(x, y, z)$ corresponding to $p_{polar}(\rho, \theta)$ is given by:

$$\begin{gathered} u = x/z = tan(\rho) \cdot cos(\theta) \ , \quad v = y/z = tan(\rho) \cdot sin(\theta) \\ \vec{L}(x, y, z) = \vec{L}(u \cdot z, v \cdot z, z) \end{gathered} \tag{7.1}$$

where z is an arbitrary scaling factor. Fig. 7.5 shows the visual field of Fig. 7.3 in the Cartesian coordinate system.

As EL, ER, and CC coincide, $\vec{L}(x, y, z)$ projects onto V_{CC} as:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = K_{CC} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = K_{CC} \begin{bmatrix} z \cdot tan(\rho) \cdot cos(\theta) \\ z \cdot tan(\rho) \cdot sin(\theta) \\ z \end{bmatrix}$$
$$i' = x'/z', \quad j' = y'/z' \tag{7.2}$$
$$p(i, j) = D_{CC}(i', j')$$

where K_{CC} is the 3×3 intrinsic matrix of camera CC, and function $D_{CC}(\cdot)$ represents distortion, which is not negligible, especially for wide-angle cameras. With Eq. 7.1 and Eq. 7.2, the mapping $(\rho, \theta) \rightarrow (i, j)$ from the human visual field (V_{EL} or V_{ER}) in the polar coordinate system to the camera visual field V_{CC} is complete. The projected visual fields for the left and right eye are denoted V_{EL}^{CC} and V_{ER}^{CC}. Fig. 7.6 shows an example of V_{EL}^{CC} and V_{ER}^{CC}.

7.4.1.2 Segmenting occlusion caused by OST-HMD

Here we address $Q2$: which part of V_{CC} is occluded by the OST-HMD. After we determine what portion of V_{CC} would normally be visible to the user, we continue to determine what area is occluded by the OST-HMD. We address $Q2$ by proposing a generic method that is able to segment the inactive area of a camera image frame. We use a pair of cameras with wide-angle lens (CL and CR) to simulate the user's eyes. We first segment the occluded area in the left and right cameras' own visual field ($O_{CL} \subseteq V_{CL}$ and $O_{CR} \subseteq V_{CR}$), and then project it to the visual field of the center camera $\left(O_{CL}^{CC} \text{ and } O_{CR}^{CC}\right)$.

Ideally, a pixel $(i, j) \in V_{CL}$ captures the frame of the OST-HMD if it is in the

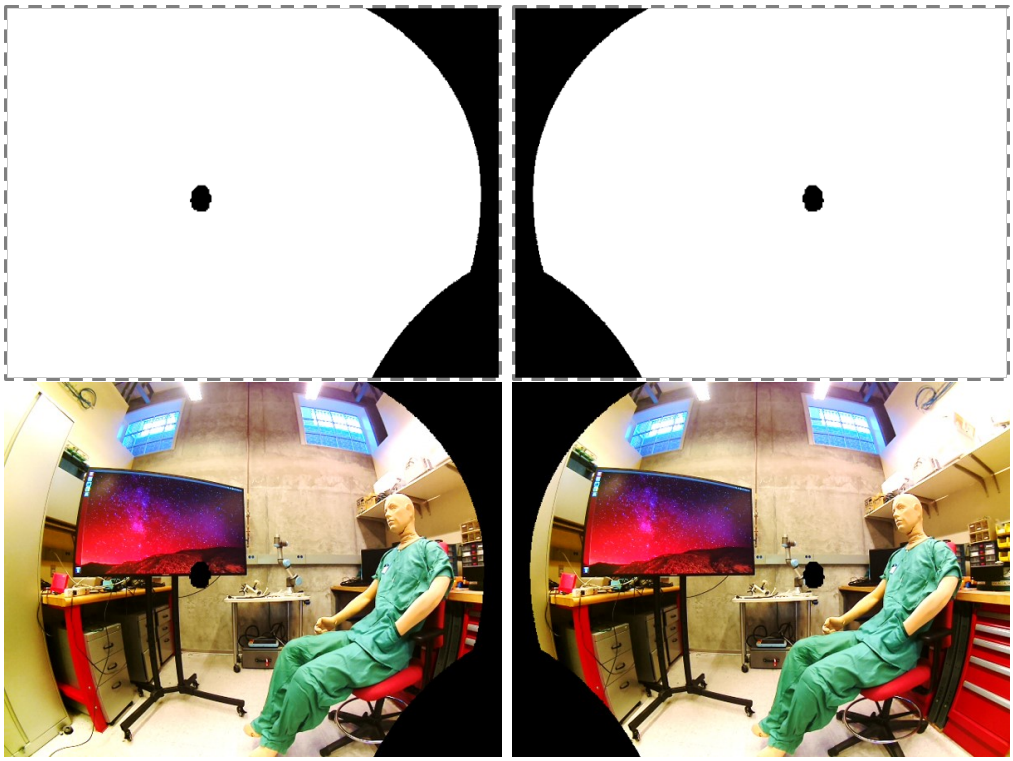


Figure 7.6: Human visual field projected on the camera visual field: V_{EL}^{CC} and V_{ER}^{CC}, demonstrated with a sample image.

occluded area, otherwise, it displays the content of the background. However, the border becomes ambiguous due to reflection and refraction caused by the optics of the OST-HMD. To resolve this ambiguity, we define a function over the image frame that finds the responsiveness score of each pixel with respect to background changes. Based on the belief that reflection or refraction area has lower responsiveness than the direct see-through area, we threshold the responsiveness scores with a threshold value T to filter out the reflection and refraction areas: $Resp_L : V_{CL} \rightarrow \mathbb{R}^+$. Overall, we define the occluded area O_{CL} as

$$O_{CL} = \{(i,j) \;\; | \;\; Resp_L(i,j) < T, \;\; (i,j) \in V_{CL}\} \tag{7.3}$$

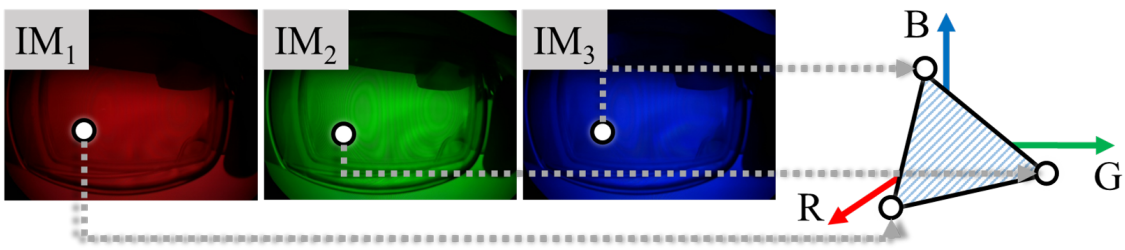


Figure 7.7: Each pixel has three (r, g, b) values in the three images of different backgrounds. The responsiveness function of the pixel is defined as the area of the triangle formed by the three color vectors.

The segmentation of occlusion is in the offline stage, so the background can be manually altered to evaluate the responsiveness of pixels. In our method, an active screen is placed at the background, and red, green, blue images are displayed sequentially on the screen. After each background image is displayed, the camera captures its current visual field, resulting in three images (IM_1, IM_2 and IM_3). For each pixel (i, j), we define the responsiveness function as the area of the triangle formed by the RGB values of the three images (illustrated in Fig. 7.7):

$$Resp_L(i,j) = \frac{1}{2}\left| \; det \begin{pmatrix} r_1 - r_3 & g_1 - g_3 & b_1 - b_3 \\ r_2 - r_3 & g_2 - g_3 & b_2 - b_3 \\ 1 & 1 & 1 \end{pmatrix} \right| \tag{7.4}$$

Next, we project the occlusion from the right and right camera's own visual field to that of the center camera, with assumptions $A1$ and $A2$. Given the camera intrinsic matrix and distortion parameters, each pixel $(i, j) \in V_{CL}$ can be mapped to a pixel $(u, v) \in V_{CC}$ by:

$$V_{CL}^{CC} = \{(u,v) \mid (u,v) = D_{CC}(K_{CC}K_{CL}^{-1}D_{CL}^{-1}(i,j)) \;,\; (i,j) \in V_{CL}\} \tag{7.5}$$

In the case where the cameras share the same intrinsic matrix and distortion parameters, Eq. 7.5 can be reduced to $V_{CL}^{CC} = V_{CL}$.

Fig. 7.11a and Fig. 7.12a show the segmentation results of HoloLens and ODG R-9. The proposed segmentation method is able to determine the occluded area, and is applicable to different OST-HMDs.

7.4.1.3 The loss of visual field

So far, we determined the portion of V_{CC} that the user normally is able to see, and what portion of V_{CC} would be invisible to the user due to the occlusion by the OST-HMD. In this section, we address $Q3$: determine the area of V_{CC} that the user is normally able to see but is not visible when wearing an OST-HMD. This is the area that the system needs to compensate for.

The visibility of a pixel $(i, j) \in V_{CC}$ is defined as:

- if $(i, j) \in V_{EL}^{CC} \cup V_{ER}^{CC}$, then it is visible to the user normally.
- if $(i, j) \in V_{EL}^{CC} \setminus O_{CL}^{CC}$, then the user's left eye can see it when wearing the optical see-through head-mounted display.
- if $(i, j) \in V_{ER}^{CC} \setminus O_{CR}^{CC}$, then the user's right eye can see it when wearing the optical see-through head-mounted display.
- if $(i, j) \in \big(V_{EL}^{CC} \setminus O_{CL}^{CC}\big) \cup \big(V_{ER}^{CC} \setminus O_{CR}^{CC}\big)$, then the user can see it with the optical see-through head-mounted display because the left or the right eye or both are able to see it).

Given the above definitions, the lost visual field when wearing an OST-HMD can

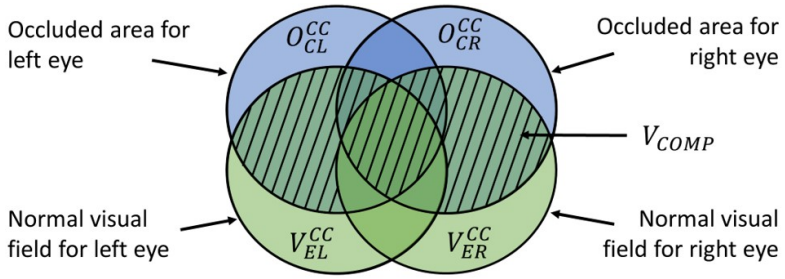


Figure 7.8: A Venn diagram for calculating V_{COMP} formulated in Eq. 7.7

be formally written as:

$$V_{COMP} = \left(V_{EL}^{CC} \cup V_{ER}^{CC}\right) \setminus \left[\left(V_{EL}^{CC} \setminus O_{CL}^{CC}\right) \cup \left(V_{ER}^{CC} \setminus O_{CR}^{CC}\right)\right] \tag{7.6}$$

In the case where the visual field of CC is smaller than the human's binocular visual field $\left(V_{EL}^{CC} \cup V_{ER}^{CC} = V_{CC}\right)$, Eq. 7.6 can be reduced to:

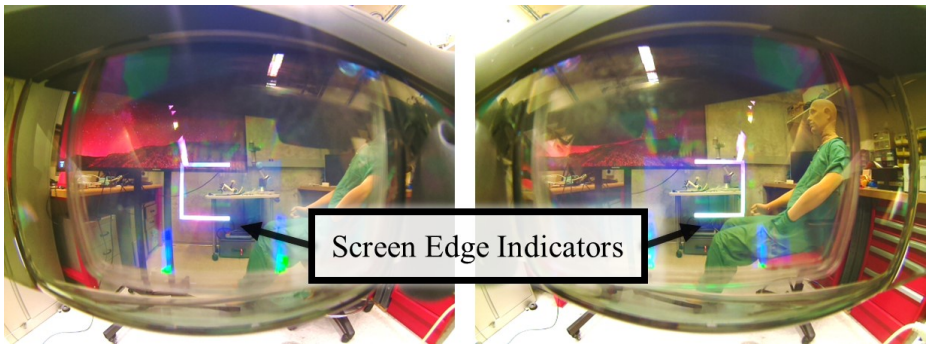
$$V_{COMP} = \left(V_{EL}^{CC} \setminus O_{CL}^{CC}\right)^{C} \cap \left(V_{ER}^{CC} \setminus O_{CR}^{CC}\right)^{C} \tag{7.7}$$

where the superscript C is the complement of a set. A Venn diagram of the above equation is shown in Fig. 7.8.

In summary, Sec. 7.4.1.1, Sec. 7.4.1.2, and Sec. 7.4.1.3 addressed $Q1$, $Q2$ and $Q3$ individually. Combining them, we are able to determine the occluded visual field when a user is wearing an OST-HMD.

7.4.2 Visualization in the Occluded Visual Field

We propose to use two types of indicator to highlight the direction of noticeable information in the occluded visual field: with the edge of the display on the OST-HMD and with an additional array of LED lights attached to the frame of the OST-HMD.



(a) Screen edge indicators on HoloLens

(b) LED indicators on HoloLens

Figure 7.9: We propose two methods to visualize the information in the occluded visual field: (a) with the edge of the display on the OST-HMD and (b) with an additional array of LED lights.

Fig. 7.9 demonstrates the user's view when they are activated with HoloLens. Each indicator has its associated occluded region of interest ($OROI$).

7.4.2.1 Screen edge indicators

Visualization with the screen edge indicators is a compact and portable solution that does not require any additional hardware. It is considered an off-screen visualization technique [86]. The screen edge indicators are the outer contour of the display, so if the user is focusing on the graphics content on the HMD, he/she is able to see these

indicators close to his/her central vision. Once the screen edge indicators are activated, the user will notice the changes and direct his/her attention to the highlighted direction. There are many pixels available for control, so that versatile information can be shown to the user. However, this method reduces the effective area for AR application. There may also exist a large gap between the location of indicators and the occluded visual field, depending on the FOV of the OST-HMD.

In this method, the edge of the screen of w pixels width is used as indicators. The edge area is discretized into N_S individual indicators. Each indicator is appearing on the visual field as $I^{CC}_{Screen,n}$, and has an $OROI$ that is denoted as $OROI^{CC}_{Screen,n}$. The $OROI$ for each indicator is a part of the entire occluded visual field.

$$\bigcup_{n=1}^{N_S} OROI^{CC}_{Screen,n} = V_{COMP} \quad , \quad \bigcap_{n=1}^{N_S} OROI^{CC}_{Scree,n} = \emptyset \tag{7.8}$$

The $OROI$s combined for all screen edge indicators cover the entire occluded visual field to be compensated for. We use the left side of the left screen to indicate information for the left half of the visual field, and the right side of the right screen to indicate the right half of the visual field. There is no overlap between multiple $OROI$s.

7.4.2.2 LED indicators

For our second method, we attached an array of LED lights to the frame of the OST-HMD, as shown in Fig. 7.9b. In [88, 295, 198], a similar setup was integrated with virtual reality or normal glasses. The LED indicators are placed at the peripheral

vision of the user so they do not interfere with the graphics content on the display and are able to closely reflect the information in the occluded area. It is "as if" the user is seeing one LED light through the hardware of the OST-HMD. If the OST-HMD does not offer an interface for custom hardware, an additional wired connection is necessary to power and control the LEDs.

The total number of LED indicators are denoted by N_L. Each LED indicator is a visualization unit that appears in the visual field at $I^{CC}_{LED,n}$, and has an associated $OROI$ which is denoted as $OROI^{CC}_{LED,n}$. There is overlap between $OROI$s. For example, when the system intends to indicate changes at the top-left direction of the user, the top-left LEDs for the left eye and right eye will both be illuminated. Tab. 7.2 summarizes the advantages and disadvantages for both types of indicators.

Table 7.2: Comparison between screen edge indicators and LED indicators

Criteria	Screen	LED
1. Number of indicators	High	Low
2. Distance between indicator and 'incident'	Far	Close
3. Interference with the content on display	Yes	No
4. Complexity of setup	Easy	Hard
5. Additional wired connection	No	Yes
6. Overlap in $OROI$s	No	Yes
7. Reflection and refraction artifacts	No	Yes

7.4.2.3 Determine OROI for indicators

Each indicator is responsible for a portion of the occluded visual field as $OROI^{CC}_{Screen,n}$ or $OROI^{CC}_{LED,n}$. The $OROI$ for each indicator is calculated by Alg. 1. This procedure

is conducted offline with cameras (CL and CR) simulating the eyes.

First, the illuminated area of the indicator $I^{CC}_{Screen,n}$ is segmented using the method described in Sec. 7.4.1.2. Each indicator is manually controlled to display red, green and blue sequentially. Because $I^{CC}_{Screen,n}$ is an area, we use the centroid of the area (a single pixel p_n) to represent each indicator. For the pixel $(i, j) \in V_{COMP}$, we find its closest indicator p_k in terms of angular distance. If this angular distance is smaller than a threshold, this pixel belongs to the $OROI$ of indicator k. The algorithm can be applied to LED indicators in the same manner as screen edge indicators. As an example, Fig. 7.11e and Fig. 7.11f show the indicators and their $OROI$s for HoloLens. Fig. 7.12c and Fig. 7.12d show the indicators and their $OROI$s for ODG R-9.

7.4.3 Information Processing of the $OROI$s

During runtime we need to determine what information to compensate for in the $OROI$s. One intuitive approach is to average the color of all pixels in $OROI^{CC}_{Screen,n}$ as the color of the indicator:

$$Color\left(I^{CC}_{Screen,n}\right) \leftarrow Avg\left(Color(i,j)\right)\ ,\ \forall (i,j) \in OROI^{CC}_{Screen,n} \tag{7.9}$$

This approach visualizes the state of the environment. It almost always assigns some color for each indicator, which might be distracting for the user. We propose to visualize the change of environment instead of the state of the environment, by calculating the optical flow [106]. The indicator is activated to display a white color when significant motion is detected in its $OROI$. The brightness value to display is

Data: Camera intrinsics and distortion parameters for CC, CL and CR. The number of indicators N_S. Threshold for angular distance θ_{thres}.
Result: The illuminated area of each indicator $I^{CC}_{Screen,n}$. The occluded region of interest for each indicator $OROI^{CC}_{Screen,n}$

begin
  Set $OROI^{CC}_{Screen,n}$ to empty for $n = 1, 2, ..., N_S$;
  for $n = 1$ **to** N_S **do**
    Display $BLACK$ in all indicators.;
    foreach *color in {R, G, B}* **do**
      Display *color* on indicator i.;
      Capture the view of camera CL and CR as $IM_{L,color}$ and $IM_{R,color}$.;
      Project $IM_{L,color}$ to center camera $IM^{CC}_{L,color}$;
      Project $IM_{R,color}$ to center camera $IM^{CC}_{R,color}$;
      Fuse the simulated binocular image $IM_{color} = (IM^{CC}_{L,color} + IM^{CC}_{R,color})/2$;
    end
    Compute the area of RGB triangle as $Resp(i, j)$ from $\{IM^{CC}_R, IM^{CC}_G, IM^{CC}_B\}$, $\forall (i, j) \in V_{CC}$;
    Threshold and segment the illuminated area $I^{CC}_{Screen,n} = \{(i, j) \mid Resp(i, j) > T,\ (i, j) \in V_{CC}\}$;
    Find the centroid of $I^{CC}_{Screen,n}$ as $p_n(i_n, j_n)$;
    The single pixel $p_n(i_n, j_n)$ represents the centroid of the n^{th} indicator.;
  end
  forall $(i, j) \in V_{COMP}$ **do**
    Compute angular distance between (i, j) and $p_n(i_n, j_n)$ as θ_n for $n = 1, 2, ..., N_S$;
    Find the smallest $\theta_k \in \{\theta_n \mid n = 1, 2, ..., N_S\}$;
    if $abs(\theta_k) < \theta_{thres}$ **then**
      Add (i, j) to $OROI^{CC}_{Screen,k}$;
    end
  end
end

Algorithm 1: The algorithm to determine the occluded region of interest ($OROI$) for each screen edge indicator

dependent on the extent of change.

Our approach resembles how human attention is attracted by stimuli in the peripheral vision. The human eye is sensitive to motion and contrast in the peripheral vision [2]. Therefore, we map the optical flow of the environment to the brightness changes of the indicators as stimuli. Another factor to consider is the egocentric motion of the user's head. When the user's head is turning, the peripheral vision is constantly changing and the human visual system is already accustomed to this. In this case, an ideal algorithm should be able to distinguish between egocentric head motion and motion of other objects at the periphery.

To detect motion we compute a dense optical flow $Flow(t)$ between the current visual field $V_{CC}(t)$ and the previous visual field $V_{CC}(t-1)$, using Gunner Farneback's algorithm [64]. The optical flow is a per pixel motion vector across the two image frames. If pixel $(i_t, j_t) \in V_{CC}(t)$ is corresponding to the pixel $(i_{t-1}, j_{t-1}) \in V_{CC}(t-1)$, then the ideal optical flow is represented as: $\overrightarrow{Flow}(i,j,t) = (i_t - i_{t-1}, j_t - j_{t-1})$.

After the optical flow is calculated, the brightness value of each indicator at the current frame is assigned to be:

$$\begin{aligned} Brightness\left(I^{CC}_{Screen,n}\right) &\leftarrow \lambda \cdot Avg\ (\ \|Flow(i,j,t)\|) \\ for \quad \|Flow(i,j,t)\| &> F_{thres}, \quad \forall (i,j) \in OROI^{CC}_{Screen,n} \end{aligned} \tag{7.10}$$

where λ is a constant coefficient that tunes the overall brightness, and F_{thres} is the threshold for minimum flow intensity to filter out noise. In addition, we added one more condition to Eq. 7.10: a simple eccentricity metric for the values of all indicators. The indicators are only activated when the maximum brightness value is significantly

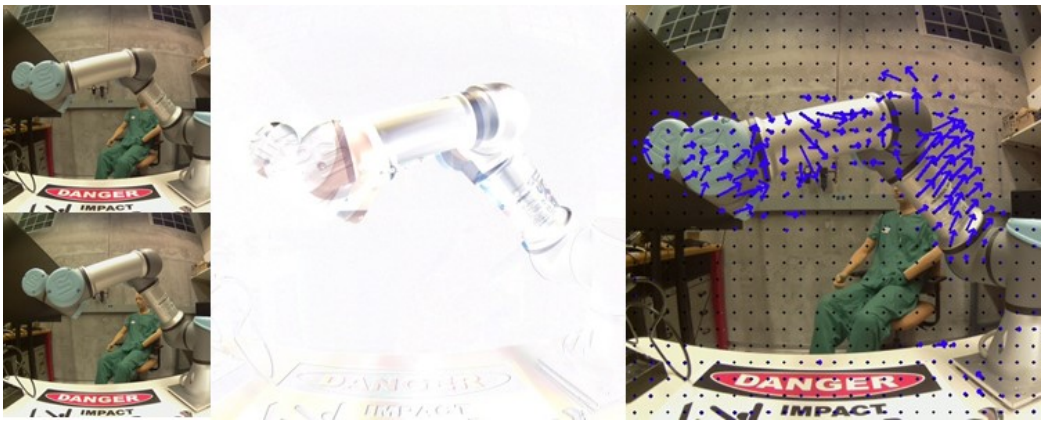


Figure 7.10: Left: the two frames captured with motion of the robot arm. Center: the absolute difference of the left images. Right: dense optical flow calculated with [64]. Each grid point has an associated arrow showing the vector of optical flow. (The scale of the vector is multiplied by 3 for better visualization.)

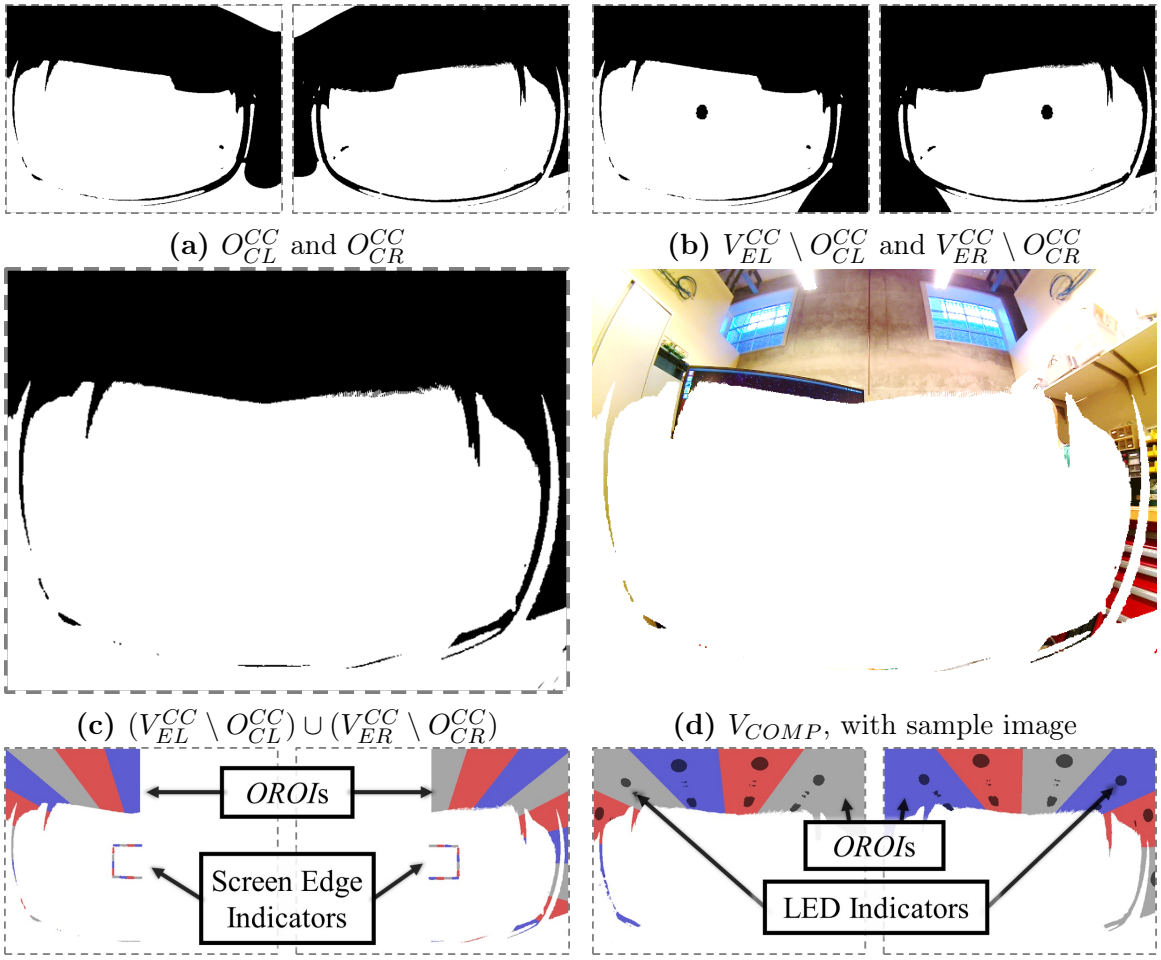
larger than the average brightness value. Otherwise, we treat it as motion of the entire frame due to the user's head motion.

With the indicators and their associated $OROI$s determined in the offline stage, and the online algorithm to process the visual field of camera CC, the real-time control loop for the indicators is complete.

7.4.4 Summary

This section addresses three questions: where, how and what to compensate for the occlusion on the user's visual field. In the offline stage, the occluded visual field and the responsible region for each visualization unit are computed. In the online stage, the optical flow at the occluded visual field is calculated in real time, and each visualization unit is controlled accordingly.

7.5 Implementation and System Setup



(a) O_{CL}^{CC} and O_{CR}^{CC}

(b) $V_{EL}^{CC} \setminus O_{CL}^{CC}$ and $V_{ER}^{CC} \setminus O_{CR}^{CC}$

(c) $(V_{EL}^{CC} \setminus O_{CL}^{CC}) \cup (V_{ER}^{CC} \setminus O_{CR}^{CC})$

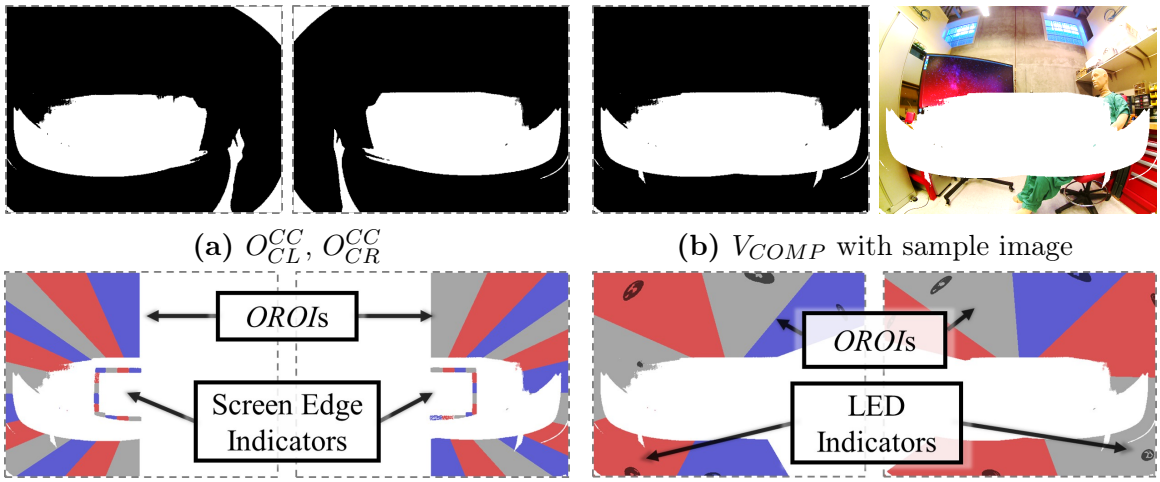
(d) V_{COMP}, with sample image

(e) $I_{Screen,n}^{CC}$ and $OROI_{Screen,n}^{CC}$ for HoloLens

(f) $I_{LED,n}^{CC}$ and $OROI_{LED,n}^{CC}$ for HoloLens

Figure 7.11: Offline stage results for Microsoft HoloLens

We integrated both screen edge indicators and LED indicators with a Microsoft HoloLens and an ODG R-9.



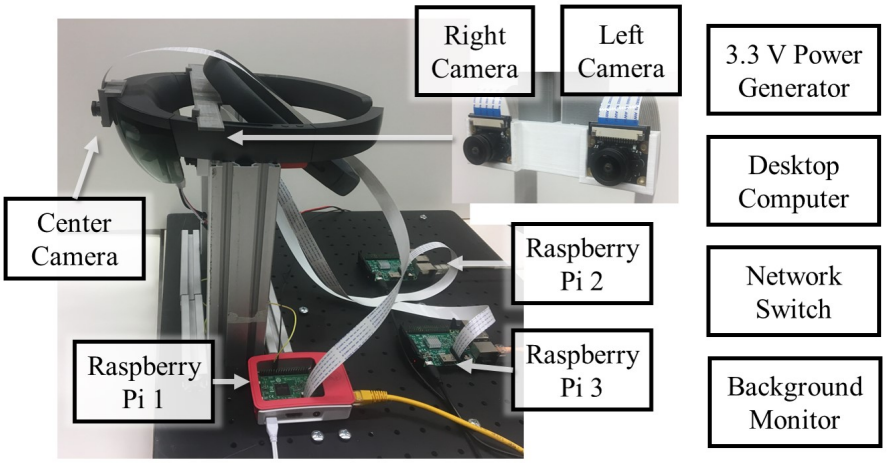
(a) O^{CC}_{CL}, O^{CC}_{CR} **(b)** V_{COMP} with sample image

(c) $I^{CC}_{Screen,n}$ and $OROI^{CC}_{Screen,n}$ for ODG R-9 **(d)** $I^{CC}_{LED,n}$ and $OROI^{CC}_{LED,n}$ for ODG R-9

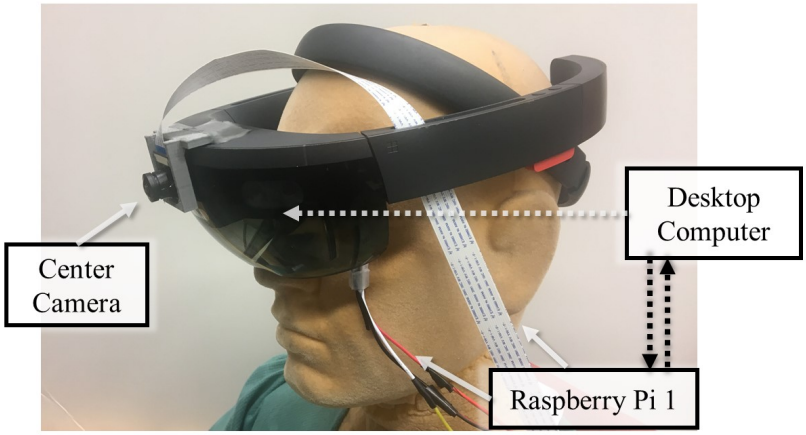
Figure 7.12: Offline stage results for ODG R-9

7.5.1 Experimental Setup for Offline Stage

Fig. 7.13a shows the system setup with Microsoft HoloLens in the offline stage. The OST-HMD is resting on a stand built from 80/20 aluminum frames. Three Raspberry Pi Model 3b are used to drive the center, left and right cameras (CC, CL, CR), which are all SainSmart 5MP Mini cameras. The eye-simulating cameras (CL and CR) are held by a 3D-printed mount with IPD of $64\,mm$. The camera holders are attached to the OST-HMD with removable glue. The Raspberry Pi that controls CC (Raspberry Pi 1) is also in charge of controlling the LED strip: Adafruit Mini Skinny NeoPixel strip of $60/m$. Three wires are required for the LEDs: control signal, GND and $+3.3V$. A desktop computer with an Intel Core $i52500@3.3GHz$ CPU and $7.7GB$ memory is responsible for the computation. The background monitor (see Fig. 7.1) is Samsung DE55A 55", with resolution 1920×1080.



(a) Microsoft HoloLens setup for offline stage



(b) Microsoft HoloLens setup for online stage

Figure 7.13: Experimental setup for offline stage and online stage.

In terms of software, the Raspberry Pis run Ubuntu Mate 16.04, and the PC runs Ubuntu 16.04 LTS. The camera videos ($640 \times 480, 15fps$) are encoded and streamed from Raspberry Pi to PC using raspivid and netcat. The exposure parameters of the cameras are controlled manually. The PC accesses the video stream via netcat, decodes the video via libx264 and FFmpeg. We use a Python library rpi_ws281x to interface the LED strip. Programs for Sec. 7.4.1.1 and Sec. 7.4.1.2 are implemented based on OpenCV 3.4. The cameras are calibrated using the OpenCV Fisheye model. The calibration results show that their horizontal FOV is 142.74° and their vertical FOV is 131.60°.

7.5.2 Experimental Setup for Online Stage

Fig. 7.13b shows the experimental setup with HoloLens for the online stage. Raspberry Pi 1 is still used to drive the center camera CC and LEDs. It streams the frames to the PC. The implementation of dense optical flow is from an OpenCV 3.4 extra module. In the method with screen edge indicators, the PC sends serialized brightness values to the OST-HMD via TCP/IP. The application on the OST-HMD includes a TCP client to receive the brightness values. The applications on both OST-HMDs are implemented with Unity. In the case where LED indicators are used, the PC sends the packet to Raspberry Pi 1 which then sets the control signal to its IO pin.

7.5.3 Microsoft HoloLens vs. ODG R-9

Figure 7.14: LED strip setup for Microsoft HoloLens (left) and ODG R-9 (right)

The setup for ODG R-9 is slightly different from Microsoft HoloLens due to the different hardware properties. Some key features and differences in their setup are listed in Tab. 7.3. Fig. 7.14 shows the LED setup for both devices. The results for offline calibration for Microsoft and ODG R-9 are shown in Fig. 7.11 and Fig. 7.12.

Table 7.3: Setup comparison for HoloLens and ODG R-9

Comparison	HoloLens	ODG R-9
Display resolution	$1268 \times 720, \times 2$	$1920 \times 1080, \times 2$
Display refresh rate	$60 fps$	$60 fps$
See-through transparency	High	Low
Number of LED indicators N_L	12	$14^\star$
Number of screen edge indicators N_S	24	24
Width of edge pixel w	50 pixels	100 pixels

7.5.4 System Performance

We measured the performance of our system. The end-to-end video streaming latency from Raspberry Pi to PC is $127\,ms$ and the per-frame computation on the PC takes $73.76\,ms$. The average framerate of the compensation loop is $13.16\,fps$.

When screen edge indicators are used, the Unity application on HoloLens runs at 32.76 fps and the application for ODG R-9 runs at 52.14 fps.

7.6 Evaluation

We conducted an objective experiment and a subjective pilot user study to evaluate the systems. In addition, we also evaluated the error introduced by assumption $A1$ (the co-location of eyes and center camera CC), and the accuracy of the segmentation algorithm of Sect. 7.4.1.2.

7.6.1 Objective Evaluation

For objective evaluation, we re-use the setup from Sect. 7.4.1.2 (Fig. 7.13a) and simulate the user's perspective with the cameras CL and CR. We evaluate four scenarios: HS (HoloLens with screen edge indicators), HL (HoloLens with LED indicators), OS (ODG with screen edge indicators) and OL (ODG with LED indicators). To present controlled motion for our objective evaluation, we display a target moving in a rectangular pattern on a monitor placed in front of the setup. Overall we observe 36 targets (6 different monitor locations $\times$ 6 targets). At each location we display a checkerboard on the monitor before the experiment begins to compute its pose relative to the OST-HMD. We show the poses and trajectories of the 36 targets in Fig. 7.15. The targets for HoloLens and ODG R-9 are not the same.

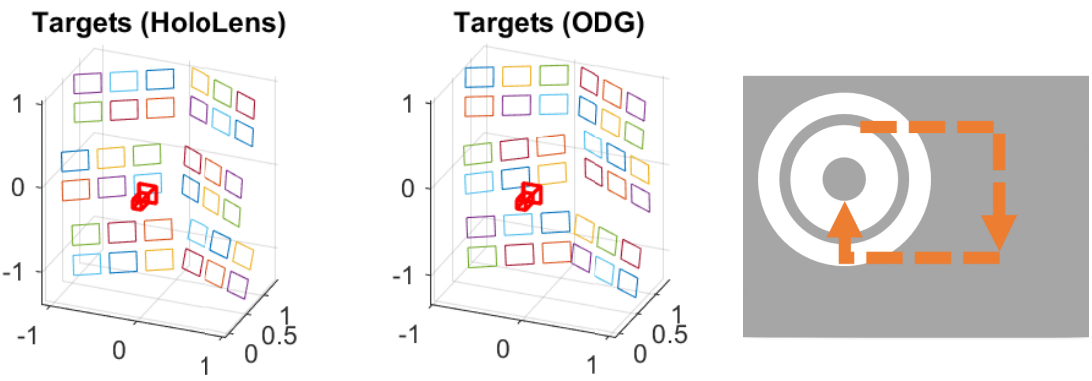


Figure 7.15: Left and middle: The pose and the trajectory of all 36 targets, for objective evaluation with HoloLens and ODG R-9. The red camera icon represents the pose of center camera CC. The units are meters. Right: The local motion of the target.

For each target, we conducted the following steps:

1. Disable both indicators, record video with CC, CL, CR. We denote them as VD_0, VD_1 and VD_2.
2. Enable screen edge indicators, record video with CL, CR. We denote them as VD_3 and VD_4.
3. Enable LED indicators, record video with CL, CR. We denote them as VD_5 and VD_6.

For each of the videos, the first image frame is subtracted from all subsequent frames in order to filter out static global luminance. Low-brightness pixels (< 20, max: 255) are thresholded to 0 for noise removal. We use VD_0 as ground truth, and fuse VD_1 and VD_2 into VD_{Eye} by alpha blending. Similarly, VD_3 and VD_4 are blended into VD_{Screen}, and VD_5 and VD_6 are blended into VD_{LED}. Then, we compute two values for VD_0, VD_{Eye}, VD_{Screen} and VD_{LED}: the average brightness value of pixels

of all frames (BR: a floating point number), and the centroid of brightness of all frames (p_{BR}: in pixel coordinates). Therefore, each video is represented by BR and p_{BR}.

We threshold the brightness value of VD_{Eye} to classify the visibility of the target into: visible, partially visible and invisible. Then, we determine if the indicators are activated by comparing the brightness values BR of VD_{Screen} and VD_{LED} with VD_{Eye}. If the indicators are activated, then the brightness value BR will be significantly higher. We summarize the success rate of our system to activate indicators in Tab. 7.4. Among the 36 targets, when the target is invisible, the system is always able to compensate for the loss of awareness for the target; when the target is visible, OS and OL both have one false positive case. OS and OL have similar performance because the $OROI$s of LED indicators for ODG R-9 also completely surround the total area-to-compensate ($\bigcup_{n=1}^{N_S} V_{Screen,n}^{CC} = \bigcup_{n=1}^{N_L} V_{LED,n}^{CC} = V_{COMP}$) . However, the HL fails to compensate in a few cases when the target is partially visible. This is because the union of all $OROI$s for LED indicators on HoloLens do not span the total V_{COMP}, e.g., the thin occluded area at the bottom.

Table 7.4: Success rate for four scenarios: HS (HoloLens with screen edge indicators), HL (HoloLens with LED indicators), OS (ODG with screen edge indicators) and OL (ODG with LED indicators)

Total (36)	HS	HL
Visible (10)	0	0
Partial (14)	14	5
Invisible (12)	12	12

Total (36)	OS	OL
Visible (5)	1	1
Partial (13)	13	13
Invisible (18)	18	18

Next, we compare the indicated direction of the target with the ground truth. Fig. 7.16 shows the calculated centroid of brightness p_{BR} for the four scenarios and 36 targets. The red circles represent the p_{BR} calculated from VD_0, as ground truth. The blue circles represent the p_{BR} calculated from VD_{Screen} and VD_{LED} when the indicators are activated. We consider these to be the perceived location of the target, which are the projection of the indicated direction on the image frame. In HS and OS, the perceived locations are closer to the image center, but with higher angular precision because there are more indicators on the screen edge than LED. For screen edge indicators, there are also cases where the blue circles are close to the red dot; this is due to the fact that when the object is partially visible, the p_{BR} is a weighted average of both the visible part of the target and the activated indicators.

Fig. 7.16 visualizes the distance between the perceived location and the ground truth. We analyze the 2D angular error between the red circles and the blue circles. The mean and standard deviation of the 2D angular errors are shown in Fig. 7.17a. The error is $4.87° \pm 5.62°$ for HS, $17.80° \pm 9.63°$ for HL, $2.99° \pm 2.34°$ for OS, and $8.07° \pm 6.08°$ for OL. With an independent two-sample t-test, we find that the 2D angular error of HS is significantly smaller than HL ($p = 1.76 \times 10^{-6}$), and that of OS is significantly smaller than OL ($p = 3.76 \times 10^{-5}$).

The 3-dimensional angular distance ($\Delta\theta$) is more related to the user's real perception of the target. We compute them by back-projecting the pixel locations into

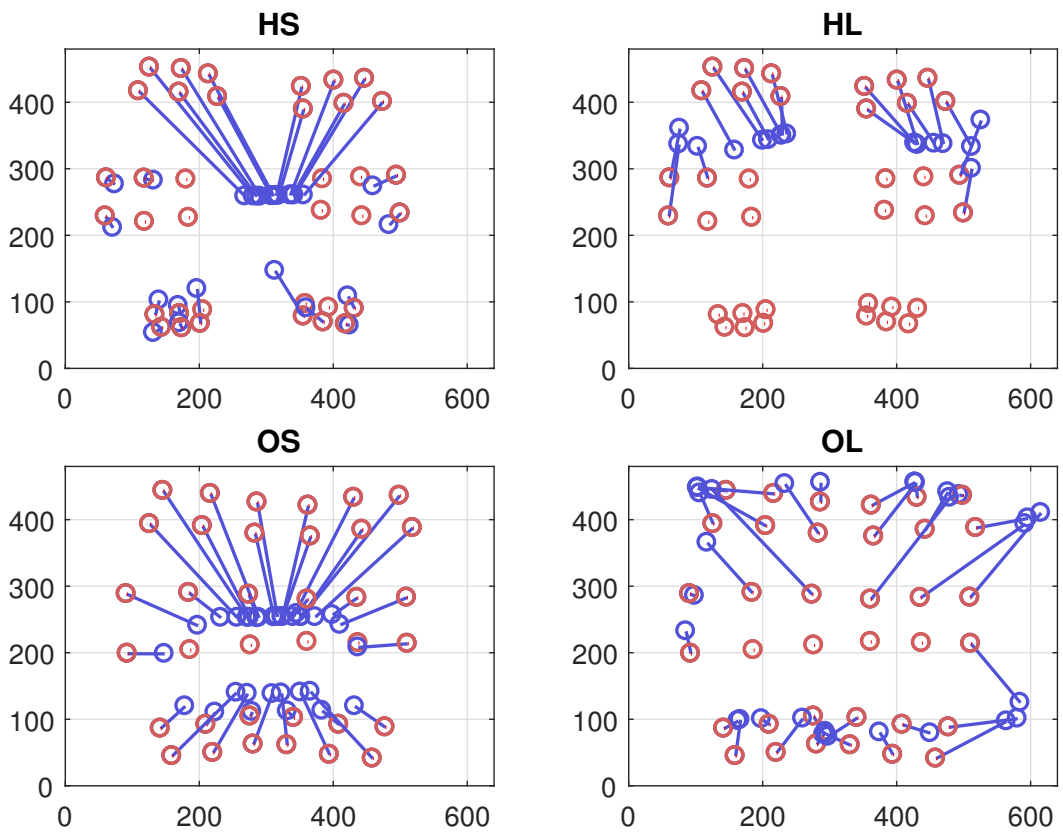
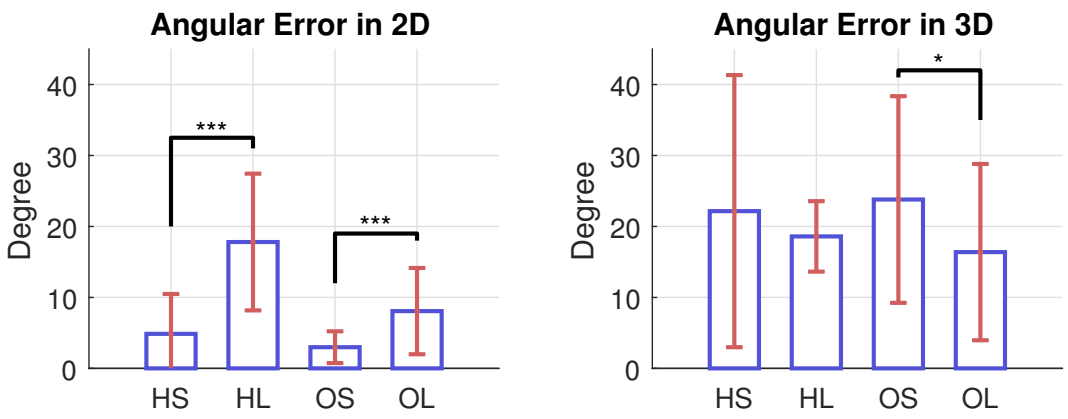


Figure 7.16: Centroid of brightness p_{BR} for all targets and the four scenarios. Red circles indicate the ground truth obtained with the center camera CC, and blue circles indicate the perceived brightness location on the image frame when the indicators are activated. (Better seen in color)

3D vectors using the camera intrinsic matrix and distortion parameters:

$$\vec{L}_{\{1,2\}} = K_{CC}^{-1} D_{CC}^{-1} \left(P_{BR\{1,2\}} \right)$$
$$\Delta\theta = \left| \; cos^{-1} \left(\frac{\vec{L_1} \cdot \vec{L_2})}{\|\vec{L_1}\| \cdot \|\vec{L_2}\|} \right) \; \right| \tag{7.11}$$

The mean and standard deviation of 3D angular error are shown in Fig. 7.17b. The error is $22.16° \pm 19.17°$ for HS, $18.60° \pm 4.97°$ for HL, $23.80° \pm 14.56°$ for OS, and $16.39° \pm 12.42°$ for OL. We assume that the distribution of the angular error is normal. With independent two-sample t-test, we find that there is no significant difference between HS and HL ($p = 0.46$), and interestingly, the 3D angular error of



(a) Statistics of 2D angular error

(b) Statistics of 3D angular error

Figure 7.17: The mean and standard deviation of angular error in 2D and 3D for the four scenarios: HS, HL, OS and OL. 2D angle is also the azimuth angle on the image plane, while the 3D angle is calculated by back-projection in Eq. 7.11.

OS is even statistically significantly larger than OL ($p = 3.23 \times 10^{-2}$). When analyzed in 3D space, the angular error of screen edge indicators becomes much larger due to the large gap between the indicators and the occluded area. There is an increase of standard deviation for HS and OS. Taking HS as an example, for occluded targets (the upper 12), P_{BR} is very close to the central vision and therefore introduces large error when analyzed in 3D. However, for partially occluded targets (e.g., the lower 12), P_{BR} is contributed by both the target itself and the indicators, and therefore appears closer to the target. The standard deviation increases drastically in consequence.

7.6.2 Pilot User Study

We conducted a pilot study of the HS and HL scenarios, with three experienced OST-HMD users, to gain more insight into the performance of our indication methods

in an actual application scenario and to get feedback on the acceptability of the two indication methods. All the users reported that they had normal vision. The synthesized normal visual field (Fig. 7.3) was used for all users.

We adopt the popular experiment paradigm of attention research: the participants are required to attend to a primary signal and a secondary signal simultaneously [2]. We implement a reading application on HoloLens where random sentences are displayed at the center of the HoloLens screen. At the same time, a background monitor is placed in front of the user, randomly showing words in random locations. The monitor is placed quite close to the user ($\sim 0.5m$) in order to cover a large FOV, and placed higher than the user because we already know that the majority of occlusion on HoloLens is the upper part. We explain the functionality of the system to the user, e.g., the flash at the screen edge and LEDs indicates the direction of a word on the background monitor. During the study, the user reads the sentences on HoloLens and also reads the words that are shown on the background monitor. We measure the success rate of the user noticing words on the background monitor. When the user is notified by our system but finds that there is no word on the background monitor, we record it as a *False Positive*.

The success rate and the number of false positives are presented in Tab. 7.5. Only 2 targets are missed for a total of 198 targets, and 5 false positives occurred. We collected feedback about their experience with the two systems and their preference. User #2 preferred HL and he mentioned that with LEDs at the periphery, he followed

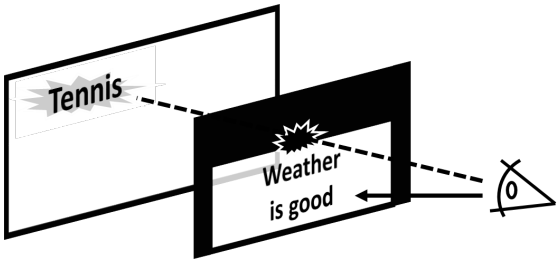


Figure 7.18: Illustration of the pilot user study on HoloLens: the user focuses on a reading task, but our system enables them to notice words on the background monitor that would otherwise be occluded by HoloLens.

Table 7.5: Results of pilot user study for two scenarios: HS (HoloLens with screen edge indicators), and HL (HoloLens with LED indicators). FP refers to the number of *False Positives* that occurred during the user study.

	HS		HL	
User	Success Rate	FP	Success Rate	FP
#1	92.6% (25/27)	1	100% (35/35)	0
#2	100% (26/26)	1	100% (31/31)	0
#3	100% (40/40)	2	100% (39/39)	1

the indication intuitively and successfully found the words, however with HS, he had to mentally calculate a direction that he should move. On the other hand, user #1 preferred HS because he thought that the LED indicators were too bright and alarming, which might be appropriate for some cases but not for the reading task. User #3 reported no preference between the two methods. User #1 also mentioned that the indicators were still activated when he was turning his head back, but he was able to understand this behavior and was adapted to it. In our current implementation, the direction of optical flow is not distinguished between moving towards the user and moving away from the user. Therefore, the system treats both kinds of motion

identically. None of the users reported uncomfortable situations.

In the pilot user study, we tested whether our systems (HS and HL) are able to correctly restore the user's awareness to look for changes (motions) in the occluded visual field. The results show that the success rate of our systems is high, despite a few false positive cases.

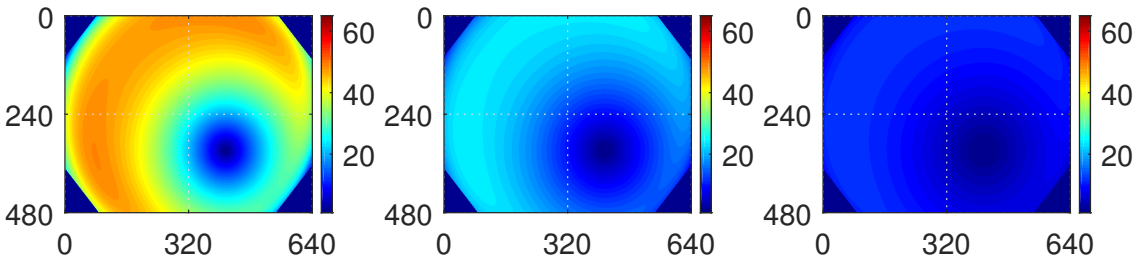
7.6.3 Co-Location Assumption

In Sec. 7.4.1, we assume that both eyes and the center camera CC are co-located ($A1$). This is equivalent to assuming that the scene is at infinity. With this assumption, we are able to project visual field of EL, ER, CL, CR to the visual field of CC without knowing the depth of every pixel. Here we evaluate the error introduced by this assumption.

For camera CL and pixel $(i_L, j_L) \in V_{CL}$, with assumption $A1$, the pixel is projected to $(i_C, j_C) \in V_{CC}$ using Eq. 7.5. To drop assumption $A1$, we denote the location of CC as $\vec{d}(x_0, y_0, z_0)$ in the coordinate system of CL. If the 3D location of pixel (i_L, j_L) is at m meters away from CL, then the pixel is projected to (i'_C, j'_C) as follows:

$$\begin{aligned} \vec{L} &= K_{CL}^{-1} D_{CL}^{-1}(i_L, j_L) \\ \vec{P} &= m \cdot \vec{L}/\|\vec{L}\| \\ (i'_C, j'_C) &= D_{CC}\left(K_{CC}(\vec{P} - \vec{d})\right) \end{aligned} \tag{7.12}$$

We use our camera calibration results of the wide-angle cameras, and assume that $d = (0.032, -0.030, 0.080)$ which is approximately the displacement between CL and



(a) $m = 0.5$ **(b)** $m = 1$ **(c)** $m = 2$

Figure 7.19: Evaluation of the error introduced by the co-location assumption $A1$. Pixel error is plotted with different pixel depths: 0.5, 1 or 2 meters. (Better seen in color)

CC in the HoloLens setup, and take $m \in \{0.5, 1, 2, 4\}$. The units are meters. We visualize the pixel distance error $\|(i_L - i'_L, j_L - j'_L)\|$ in Fig. 7.19. The corners of the image frame are excluded due to inconsistent behavior of the *cv::fisheye::distortPoints()* function in OpenCV. The mean and standard deviation of the pixel error introduced by the co-location assumption are 36.05 ± 10.71 pixels, 17.51 ± 5.48 pixels, 8.61 ± 2.77 pixels, and 4.26 ± 1.39 pixels for pixel depths at $0.5m$, $1m$, $2m$ and $4m$, respectively. The error introduced by assumption $A1$ decreases when the pixel depth increases.

The displacement between the virtual eye and real eye causes perceptual issues with a VST-HMD [26]. In our setup with an OST-HMD, the displacement does not directly affect the major part of the user's vision. However, the displacement still introduces error to the desired direction of indication. To completely eliminate assumption $A1$, it is necessary to determine the position of the eyes, e.g., by [196], and the pixel depth in real time, either by RGBD sensors or by software reconstruction methods like [51].

7.6.4 Segmentation with Responsiveness Function

In order to evaluate our segmentation algorithm, we manually segment the occluded area by HoloLens as ground truth. Fig. 7.11a shows the segmentation results of our algorithm, while Fig. 7.20a and Fig. 7.20b show the corresponding ground truth. Note that in the manual segmentation, all relative scotoma are removed, e.g., the sharp angles on the glass surface. Our algorithm is dependent on the threshold T of the responsiveness function $Resp(\cdot)$ of all pixels. The area that should be part of the occlusion but is not correctly segmented is *False Negative*. The area that should not be part of the occlusion but is wrongly segmented is *False Positive*.

Fig. 7.20c and Fig. 7.20d show the percentage of false positive and false negative for the segmentation of HoloLens. The minimum combined error rate is 2.99% for O_{CL}^{CC} and 5.66% for O_{CR}^{CC}. A threshold of $240 \leq T \leq 4170$ will be able to generate segmentation results with a combined error rate of less than 10% for both images.

The performance of our system is highly dependent on the segmentation result: O_{CL}^{CC} and O_{CR}^{CC}. With a high false positive rate, the system will indicate the motion that is already naturally seen by the user, which results in a more alarming or more disturbing system, depending on the application. A segmentation with a high false negative rate will fail to indicate certain activity that happens in the wrongly segmented area. Alternative algorithms, e.g. morphological filter [264] and watershed transformation [217], can be applied for segmentation as well.

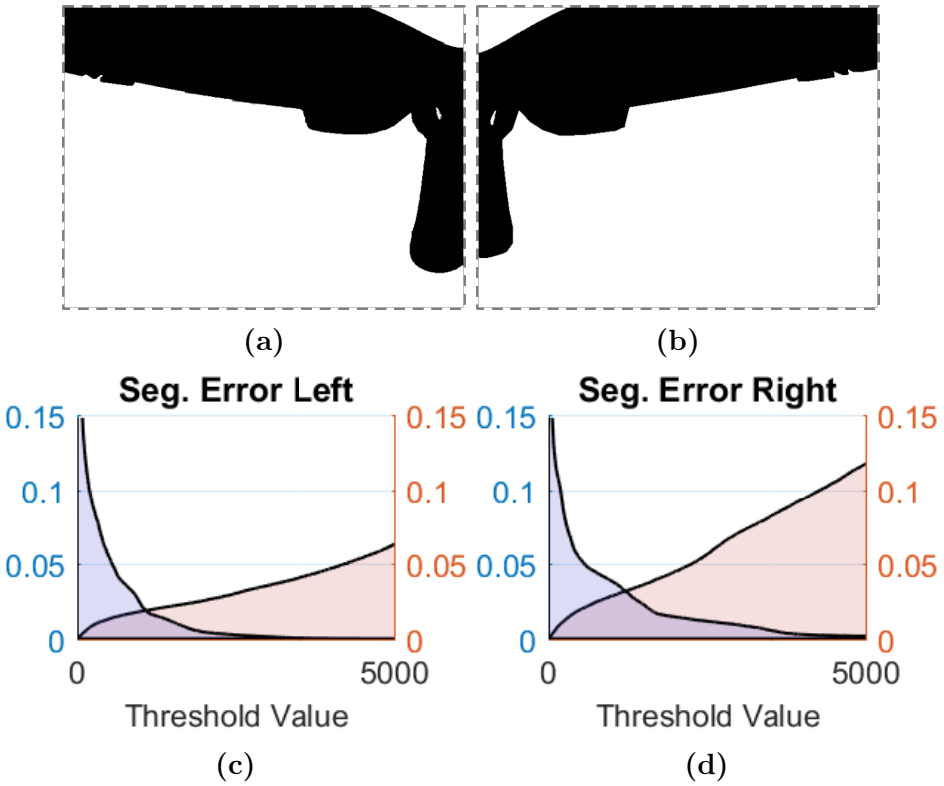


Figure 7.20: (a)(b): Manual segmentation of O_{CL}^{CC} and O_{CR}^{CC} (of Fig. 7.11a) for HoloLens, used as ground truth. (c)(d): The evaluation results of (a)(b). Horizontal axis is the threshold T for the responsiveness function. Left vertical axis: false positive rate. Right vertical axis: false negative rate.

7.7 Discussion

7.7.1 Screen Edge and LED Indicators

From the evaluations of our prototype, the screen edge indicator and LED indicator both yield good success rate, but show distinct characteristics. The number of screen edge indicators is more than the number of LEDs for HoloLens, leading to fewer number of pixels in $OROI$ in average. As a result, Eq. 7.10 yields a higher value and therefore makes the screen edge indicators more 'sensitive' than LED in-

dicators. The false positive results (14 versus 5 in objective evaluation, 4 versus 1 in user study) match the above hypothesis. A more 'sensitive' indication will benefit applications that involve less frequent but critically hazardous events, e.g., collaborative manufacturing with a robot. As mentioned by user #2, the LEDs placed at the periphery provide intuitive indications on HoloLens. On the contrary, limited by the FOV of HoloLens, the screen edge indicators are much closer to the central vision, and thus introduce a larger 3D angular error. The locations of both indicators are highly dependent on the FOV and structure of the OST-HMD.

In conclusion, the choice between the two indicating methods should be decided considering the specific application and in the context of a specific OST-HMD. In this chapter, two indication methods are realized and evaluated, but a more thorough user evaluation with more attention guidance methods in a specific application is necessary in the future.

7.7.2 Expansion of the Awareness

In fact, if the camera CC has a larger visual field than the human binocular visual field $V_{EL}^{CC} \cup V_{ER}^{CC} \subset V_{CC}$, it is possible to expand the normal human visual experience instead of only compensating for the occlusion: $V_{EXPAND} = V_{CC} \setminus \left[\left(V_{EL}^{CC} \setminus O_{CL}^{CC}\right) \cup \left(V_{ER}^{CC} \setminus O_{CR}^{CC}\right)\right]$. Note that when the center camera CC has a FOV larger than 180°, the projection matrix is not able to cover the entire FOV. In this case, it is more suitable to represent the visual fields in the polar coordinate system

where ρ could be larger than 90°.

7.7.3 Personalized Visual Field

The human visual system is complex and a person's visual field is dependent on the age and health of the eyes. The measurement of the human visual field is also dependent on the size, color and luminance of the target [220]. The nominal human visual fields we use in the experiments are synthesized with typical features of normal human visual fields. A mismatch between the nominal visual field and actual visual field can cause false positives or false negatives. If a measured visual field of the user is available, it can be seamlessly incorporated into our workflow by substituting it for the nominal V_{EL} and V_{ER}, thereby avoiding these issues.

7.7.4 Optimization for Implementation

In our current prototype, the latency for video streaming and computation is $127\,ms$ and $73.76\,ms$ respectively, which is not ideal for real-life applications, especially in potentially dangerous situations. The streaming latency can be reduced by using hardware accelerated codecs, or even be eliminated by migrating the computation to the Raspberry Pi or HMD onboard processor. The computation can be accelerated with the support of a GPU. A camera with higher shutter speed will also help reduce the overall latency.

The Raspberry Pi and wirelessly connected PC limit the mobility of the user because both HoloLens and ODG R-9 are untethered devices. This design choice is partly due to the lack of custom hardware interface of current OST-HMDs. Ideally, if the OST-HMD offers a wide-angle camera, a wide field-of-view for display (or periphery display for off-screen visualization), and sufficient computational power, the restoration of awareness can be enabled purely with software. It is our hope that future OST-HMD manufacturers became aware of the potential danger and provide such interfaces.

In terms of the algorithm for online processing, we use optical flow to parse the information in the $OROI$s. More context understanding can be built into our framework by substituting the online image processing algorithm, such as egocentric object tracking [6] or SLAM++ [223], depending on the specific requirement for applications.

7.8 Conclusion

In this chapter, we investigate the occlusion of peripheral vision caused by the current generation of OST-HMDs. The occlusion causes partial loss of peripheral vision, however, peripheral vision is essential for mobility and safety. We calculate the area that is occluded by the OST-HMD in an offline calibration stage and match it with the image captured by a wide-angle scene camera. We detect motion in the occluded area through an optical flow algorithm. We propose two methods to indicate

the detected motion to the user to restore his/her awareness of the surroundings.

We implement our methods on state-of-art OST-HMDs. Four prototype systems are evaluated and compared: two OST-HMDs (Microsoft HoloLens and ODG R-9) with two types of indicators (screen edge indicators and LED indicators). In the objective evaluation, 36 targets are tested for each of the prototypes. The success rate of the systems to activate in case of occlusion is 100%. The false positive rate is 6.67%. Then, we analyze the 2D and 3D angular error of the indicated direction compared with the ground truth. Screen edge indicators have significantly smaller 2D angular error than LED indicators. However, the performance of the two indicators in 3D angular error is similar. We carried out a pilot user study in which our system had a success rate of 98.90%, despite 5 false positive cases. Users have different preferences between the two types of indicators. In addition, we also evaluate the assumption of this chapter and the segmentation method for determining the occlusion.

The results show that both methods are viable for compensating the loss of safety-critical information in the occluded areas. Our vision is that designers of future OST-HMDs will try to alleviate the occlusion problem by using more see-through materials, or provide see-through ability using our methods or even LCD panels.

In surgical applications, which are the main focuses of this thesis, OST-HMD occlusion may be less of a safety issue, but can affect the situation awareness of the surgeon. One advantage to modeling the human visual field is that in a digitally-integrated OR, it becomes possible for the system to identify what the surgeon cannot

see and take appropriate action. For example, if there is important information on a display outside the surgeon's field of view (due to OST-HMD occlusion or other factors), the indicator can be used to alert the surgeon. In some cases, this may be a better than displaying the information directly on the HMD, which could interfere with the surgeon's view of the patient.

7.9 Closing Remarks

The ultimate OST-HMD will be similar to normal glasses that do not cause any occlusion. At that time, our method for restoring the awareness will hopefully be retired, and possibly find usage to expand the awareness (beyond the normal field of view) or in a heads-up display integrated into a car, e.g., due to the occlusion caused by the A-pillar. However, before the ultimate goal is achieved, the occlusion caused by the current generation of OST-HMDs is still a common issue. Our vision is that, in the near future, OST-HMD manufacturers will use more see-through materials for the hardware structure, or even embed micro displays where the occlusion is unavoidable, and that AR application designers will become more aware of the potential danger in the occluded visual field and make the applications safer for users.

7.10 Published Work

Material from this chapter appears in the following publication:

1. **Long Qian**, Alexander Plopski, Nassir Navab, Peter Kazanzides, "Restoring the Awareness in the Occluded Visual Field for Optical See-Through Head-Mounted Displays," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, Volume 24, Issue 11, pp. 2936-2946. IEEE. 2018.

Chapter 8

Summary and Conclusions

8.1 Summary of Chapters

This dissertation focuses on applying an optical see-through head-mounted display (OST-HMD) in computer-integrated interventions. OST-HMD, as a new and popular medium for augmented reality (AR), can provide high-quality augmentation over the user's natural field-of-vision, which has huge potential to be a useful aid during clinical procedures. Chapter 1 presents the current state-of-art of the hardware of OST-HMD, and the challenges to apply an OST-HMD in a clinical setting. We contributed to this topic via technical contributions and clinical contributions.

Chapter 2 introduces the importance of display calibration for accurate augmentation, and presents our innovative methods to improve the display calibration procedure. The display calibration is fundamental to surgical guidance applications.

The misalignment between the reality (patient anatomy) and the virtuality (guidance information) is critical in a clinical environment. In this chapter, we first present fh-SPAAM, which mainly limits the user interaction space when performing the alignment task, and therefore reduces the error introduced by the user. The evaluation results showed that the user alignment error is significantly reduced via our method. Apart from fh-SPAAM, we proposed another improvement to the traditional SPAAM methods which takes into account the physical constraints of a stereoscopic OST-HMD, e.g. the fixed distance between the eyes. These constraints regulate the optimization problem. In the third methods introduced in this chapter, we model the complex display system and rendering pipeline of stereo OST-HMD as a "blackbox", and transform the display calibration into a 3D-3D registration problem. The method can be easily integrated with the current generation of OST-HMDs and popular development platforms of AR. It is deployed with different OST-HMDs (Microsoft HoloLens v1 and Epson Moverio BT-300), and with different tracking systems (head-anchored and world-anchored). The "blackbox" approach is mainly applied in the AR applications of later chapters.

Chapter 3 introduces another technical contribution to increase visual acuity of an OST-HMD, both for the reality and for the virtuality. We developed AR-Loupe, a prototype to validate the concept of zoomable augmented reality. AR-Loupe is developed by integrating an OST-HMD (Magic Leap One) with optical loupes. A dedicated two-step system calibration procedure is designed to align the virtuality

with the reality across the user's field of vision. We compare AR-Loupe with a baseline OST-HMD setup for the guidance accuracy. The results showed that the users were able to achieve sub-millimeter operation accuracy ($0.82\,mm$) with the increased visual acuity using AR-Loupe. The improvement is significant ($p = 1.38 \times 10^{-7}$) compared to the baseline setup ($1.49\,mm$). AR-Loupe has the potential to guide operations where high visual acuity is required, e.g., dental or micro-surgery.

Chapter 4 describes an OST-HMD-based AR application for image-guided surgery, the "Virtual Monitor". In image-guided surgery, the medical image is generally displayed on stationary monitors that are not co-located with the patient and the line-of-sight could be blocked during the procedure. The virtual monitor can potentially solve the ergonomics problem using an OST-HMD. The medical images visualized on the virtual monitor can be anchored to the physical world or as a heads-up display for the surgeon. We proved the feasibility of the virtual monitor in phantom-based percutaneous spine procedures in a clinical environment. This novel visualization approach may serve as a valuable adjunct tool during minimally invasive percutaneous spine treatment. Furthermore, with more OST-HMD devices coming into the market, we propose a set of criteria to evaluate different OST-HMDs for potentially being used for virtual monitor applications. The criteria include text readability, contrast perception, task load, frame rate and system lag. This chapter introduces our first clinical use case to explore the use of OST-HMD for surgical interventions.

Chapter 5 introduces *ARssist*, which is another OST-HMD-based AR application,

targeting at the patient-side assistant in robotic-assisted surgery. The teleoperated surgical robots, e.g. the da Vinci robot, bring significant advantage to the console surgeons but do not improve the ergonomics of the patient-side assistants. However, the assistants play a critical role to the success of the robotic-assisted surgeries. *ARssist* integrates an OST-HMD with the da Vinci robot. It uses the virtual monitor presented in the previous chapter to visualize the laparoscopy for the bedside assistant, and displays the "hidden" robotic instruments, robotic-driven endoscope and its frustum inside the patient body, corresponding to their physical locations. We evaluated *ARssist* at multiple sites (the Chinese University of Hong Kong, Johns Hopkins University and Intuitive Surgical Inc.) along with system iterations between the evaluations. Two frequent tasks of the assistants are evaluated in a phantom setup: instrument insertion and tool manipulation. For inexperienced users, *ARssist* provides significant improvement in all objective task performance metrics, and subjective metrics, e.g. the hand-eye coordination. For experienced users, *ARssist* provides useful guidance that significantly improves the trajectory deviation during instrument insertion. Other objective metrics show improvement but are not significant via statistical analysis. Their subjective feedback is significantly positive with *ARssist*. These results indicate that *ARssist* may be a useful surgical guidance aid for the patient-side assistant. The clinical benefits need to be further assessed with clinical studies.

Chapter 6 introduces *ARAMIS*, an OST-HMD-based AR application targeting at

laparoscopic surgery. *ARAMIS* demonstrates the concept of "see-through surgery", where the surgeons can see 3D real-time patient internal anatomy, as if in an open procedure. *ARAMIS* aims to bring back the ergonomics of open surgery to minimally-invasive surgery, providing natural hand-eye coordination and depth perception to operate laparoscopic instruments. From the technical standpoint, the surgical site captured by the stereo laparoscope is 3D reconstructed as a point cloud, and wirelessly streamed to HoloLens for in-situ visualization. The reconstruction and streaming are optimized for bandwidth and latency, so that the point cloud ($117k$ points) is refreshed at $41.27\,Hz$ with a latency of $158.7\,ms$. The system performance enabled us to evaluate *ARAMIS* in a peg transfer study, comparing the failure cases, completion time, and subjective feedback with a traditional laparoscopic setup. With *ARAMIS*, the number of failure cases is dropped from 11 to 8. The completion time is similar on average, but shows improvement when the laparoscope and the user have relatively large mis-orientation. The subjective feedback shows that the users prefers *ARAMIS* in terms of the hand-eye coordination and confidence of operation.

Chapter 7 investigates a technical challenge with the current generation of OST-HMDs, which is the occlusion caused by the hardware. The hardware frame of the OST-HMD generally occupies part of the user's peripheral vision, which is potentially dangerous. In a surgical setting, the occlusion at the periphery will impact the surgeon's situation awareness. We model the occlusion problem and describe our method to alleviate the issue. We propose to use a wide field-of-view camera to first capture

the environment, then extract the motion in the background using optical flow algorithms, and map the extent of motion to the indicators distributed at the periphery (LED lights attached to the OST-HMD or the edge of the OST-HMD screen). We developed prototypes for HoloLens and ODG R-9, and evaluated them both objectively and subjectively. The evaluation validated our method of modelling the occlusion issue with OST-HMD, and proved that both LED and screen edge indicators are able to cover the lost awareness of the environment.

In the appendices of this thesis, we present a comprehensive review of AR applications in the field-of robotic-assisted surgery, compilation guides on *ARssist* and *ARAMIS*, and a programming guide to fellow researchers who are interested in developing AR applications on OST-HMDs using Unity.

8.2 Conclusion

An OST-HMD offers additional information display overlaid on the user's normal vision. There is a growing interest to apply OST-HMD for AR applications in the clinical environment. However, there are many technical challenges and clinical challenges that need to be addressed before the actual clinical use. The thesis contributes to the community with novel methods to overcome some existing technical challenges, e.g.: i) the difficulty of offering accurate alignment between the virtual and the real, ii) the awareness lost at the periphery, and iii) the lack of capability to increase visual acuity

with an OST-HMD. With the current generation of OST-HMDs, we are at the point that specific clinical applications can be developed and evaluated, which will serve to facilitate further refinement and future deployment. This thesis contributes by exploring some clinical use cases where an OST-HMD offers perceptual benefits, e.g.: i) virtual monitor for image-guided surgery, ii) *ARssist* for robotic assisted surgery and iii) *ARAMIS* for minimally-invasive surgery. We evaluate these prototypes in preclinical studies to assess the impact of the AR application on task performance and ergonomic factors. The evaluation showed that the visualization of hidden robotic instruments and endoscope in *ARssist* improves task safety, hand-eye coordination for the experienced users, and it significantly improves task performance for inexperienced users, suggesting that *ARssist* can ease the learning curve for bedside assistants. For *ARAMIS*, the users reported significantly improved hand-eye coordination and depth perception, and it makes laparoscopic instrument manipulation easier when there is significant mis-orientation of the laparoscopic video. The subjective feedback strongly favors *ARAMIS* in terms of the confidence level. Overall, we have demonstrated that OST-HMD based AR application provides ergonomic improvements, e.g. hand-eye coordination, and in challenging situations, the improvements in ergonomic factors lead to improvements in task performance.

8.3 Future Work

Many other technical challenges need to be investigated and addressed in the future, e.g., the field-of-view, size, weight and form-factor of the OST-HMD. Throughout this thesis, we have built a few prototypes based on existing OST-HMDs, e.g., the AR-Loupe. In the future, the concepts of the prototypes should be built into the specialized hardware.

With increasing computational power onboard, powerful algorithms, e.g. deep learning, can be implemented on the headset to improve the capability to process information. The AR applications in this dissertation aim to provide task-specific assistance to the user. Therefore, the algorithms should be relatively simple and fast in order to provide such feedback in real-time. In the future, an OST-HMD with powerful algorithms, or "intelligence", could be viewed as an intelligent assistant to the surgeon, being able to recognize the environment, the current clinical task and the intention of the surgeon.

In this dissertation, we have conducted many multi-user studies to evaluate our proposed applications. Most of them are phantom studies instead of ex-vivo or in-vivo results. Although the preliminary experiments have demonstrated the feasibility and benefits, much more efforts are needed to evaluate the actual clinical impacts in a clinical evaluation setting.

In the near future, I think AR applications such as the virtual monitor will see some initial deployment in the operating room. The virtual monitor does not signif-

icantly change the current surgical workflow. It provides an alternative way to see medical images that are currently displayed on 2D monitors in the OR. In fact, a few products that adopt this concept have been approved by the FDA for clinical use. I think large-scale multi-site clinical studies with follow-ups need to be done, to facilitate the understanding of the community, e.g.: i) the clinical impact at different stages of the treatment, ii) the cost of such AR products, and iii) whether the impact, if positive, justifies the cost of the system.

8.4 Closing Remarks

We summarize the top-three positive impacts of AR as an assistance for surgical interventions as:

1. improved hand-eye coordination with co-located augmentation
2. situation awareness improvement with registered pre-op plan or pre-op model (tumor location)
3. an ideal interface for an "intelligent assistant" that understands the procedure and provides useful guidance based on the "understanding".

This thesis has provided contributions in some of the above areas. We also summarize the top-three challenges that we recommend fellow researchers to address in order to realize the positive impacts of AR:

1. large-scale multi-site clinical studies with follow up to understand everything

about the new "product" in the operating room

2. customized OST-HMD that caters to specific clinical requirements
3. explore more clinical use cases, including surgical ones and non-surgical ones, e.g. patient education, remote medicine.

With continuous efforts to overcome the technical and clinical challenges, we believe the potential of AR and OST-HMD in the operating room will be fully revealed in the future.

Appendix A

A Review of AR-Integrated RAS

Augmented Reality (AR) and robotic-assisted surgery (RAS) are both rapidly evolving technologies in recent years. RAS systems, such as the da Vinci® Surgical System, aim to improve surgical precision and dexterity, as well as access to minimally-invasive procedures, while AR provides an advanced interface to enhance user perception. Combining the features of both, AR-integrated RAS has become an appealing concept with increased interest among the academic community. In this appendix, we provide a comprehensive review of the existing literature about AR-integrated RAS.

Robotic-assisted surgery (RAS) has tremendously shifted the way that many surgeons operate. Surgical robots are able to improve precision, access to critical anatomy, as well as surgeon autonomy and ergonomics [255, 32]. The most successful commercial surgical robot so far is the da Vinci® Surgical System [90] developed by

Intuitive Surgical Inc. (Sunnyvale, CA). To date, it has been used for more than 6 million cases around the world. Despite their clear clinical benefits, the lack of tactile feedback, extra training requirements and cost have been viewed as major limitations of the current generation of medical robots [136, 259, 186, 190].

On the other hand, AR is essentially an advanced user interface, in which additional information can be superimposed on the operator's view of a scene. It can potentially enhance the surgeon's perceptual abilities.

The earliest papers that detailed the combination of AR and RAS were published in 2001 by Wörn et al. [292] and Devernay et al. [56]. Wörn et al. integrated KasOp, an operation planning system, with the CASPAR (Orto Maquet, Germany) orthopaedic robot, for craniofacial surgery. During the surgery, the planning data (bore holes, intersection and osteotomy lines, etc.) could be projected onto the patient's head [292]. Devernay et al. identified the difficulty of distinguishing two coronary arteries in robotic-assisted cardiac surgery due to the narrow field-of-view through the laparoscope. The authors proposed to use AR to enhance the surgeon's situation awareness [56].

Since then, researchers have published many inspiring concepts, innovative methods, technical solutions or clinical studies in the domain of AR-integrated RAS. At this time, both AR and RAS technologies have become well recognized and developed to a certain extent. Therefore it is timely to perform an interdisciplinary survey of the existing literature combining the two domains, to give researchers an overview of

the current status of AR-integrated RAS.

A.1 Review Methods

Paper Search

We collected 154 papers from Scopus and 108 papers from PubMed databases (as of May 5, 2019) by searching for the following keywords in the title and the abstract: *("augmented reality" OR "mixed reality") AND ("medical robot" OR "surgical robot" OR "medical robotics" OR "da vinci" OR "surgical robotics" OR "robot assisted" OR "robotic assisted" OR "robotic aided" OR "robot aided" OR "robotic surgery")* [1]. We used various synonyms in order to include as many relevant papers as possible. We first excluded papers that were: 1) not written in English, 2) duplicated existing work, 3) irrelevant, 4) unavailable, and 5) non-original, e.g., review papers. 15 additional papers were identified to be relevant, via cross-referencing or other sources, and were manually added to the collection. In total, 93 papers are included in this survey. [174, 292, 56, 173, 100, 46, 41, 271, 62, 141, 5, 272, 140, 243, 270, 68, 161, 4, 246, 293, 297, 240, 285, 281, 283, 282, 284, 280, 249, 40, 267, 81, 15, 36, 202, 296, 277, 3, 226, 266, 268, 278, 232, 22, 239, 265, 131, 204, 203, 167, 165, 166, 7, 146, 160, 205, 96, 92, 143, 147, 71, 57, 58, 184, 144, 262, 79, 116, 24, 183, 302, 227, 305, 42, 191, 304, 233,

[1] The keyword *"robotic assisted"* was used to cover papers like *"robotic assisted prostatectomy"*, but also included irrelevant papers such as *"robotic assisted maintenance"* which were filtered out in the next step.

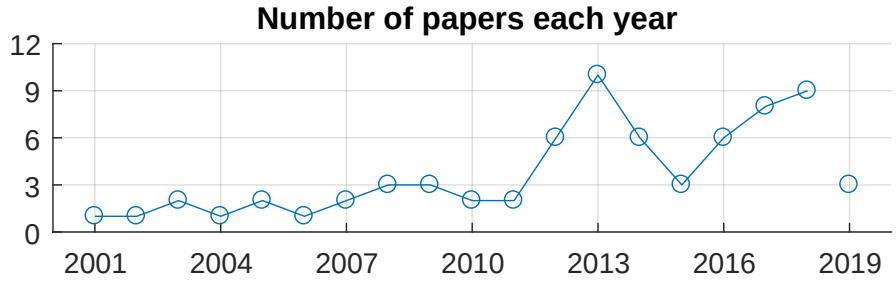


Figure A.1: The number of publications each year about AR-integrated RAS that are included in our survey (as of May 5, 2019).

126, 307, 129, 303, 207, 199, 20, 59, 201, 236, 138, 109, 224, 80, 108, 200].

Inclusion of AR-Ready Papers

During the screening of papers, we noticed that there is a discrepancy among researchers regarding the definition of AR. We have previously mentioned our definition of AR in Sect. 1.1 which follows the widely accepted definition of AR by Milgram et al. [163] and Azuma et al. [16]. We exclude VR and AV papers, and self-proclaimed "AR" systems that simply display robotic laparoscopy side-by-side with medical images, without any attempt to register different modalities.

However, there are some papers that orient the anatomy model so that it appears aligned with the laparoscopic video, but the two views are not directly overlaid due to technical issues or clinical concerns. In this case, a subset of degrees of freedom is registered. We consider such papers as **AR-ready** because they are very close to an actual AR implementation, but are not yet strictly AR by definition. There are 8 AR-ready papers in total [266, 268, 265, 20, 201, 167, 165, 166]. Fig. A.3 shows an

example of an AR-ready system. We decided to include these AR-ready papers in our review because they present interesting use cases and valuable clinical findings.

Paper Analysis Strategy

We classify the relevant papers into three main categories: *Algorithm* (32), *Hardware Tool* (6), and *Application* (71) [2]. More specifically, papers in the *Algorithm* category present novel software algorithms to address technical problems, and papers in the *Hardware Tool* category present the development of some specific hardware instruments to facilitate AR-integrated RAS, e.g., an endoscopic structured light system [143]. Our survey mainly focuses on the papers in the *Application* category, where AR is applied in certain procedures of RAS, e.g., intraoperative guidance. The taxonomy of paper classification is illustrated in Fig. A.2.

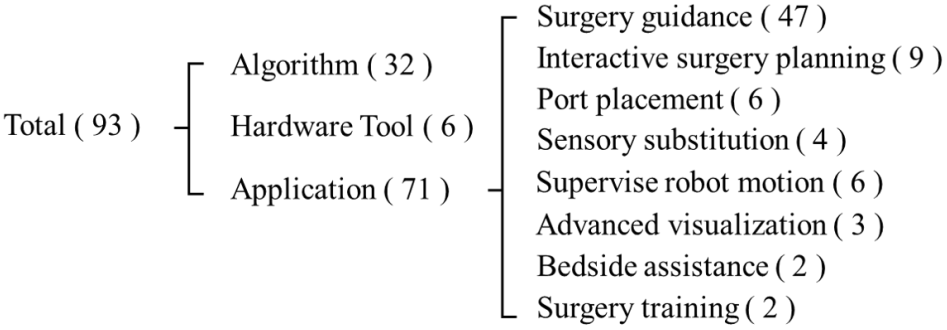


Figure A.2: Classification of literature about AR application in RAS

We also analyze the meta information of the papers with focus on AR application in RAS, specifically, the year of publication and number of citations from Google

[2] A paper that presents both a novel algorithm and an application is counted in both categories. A paper may also present more than one application paradigm of AR-integrated RAS.

Scholar (as of May 5, 2019). Fig. A.1 shows the number of publications per year. The papers are, on average, cited 3.30 times per year.

A.2 Types of Medical Robot

We record the types of robotic systems used among the reviewed literature, categorize them as shown in Tab. A.1, and discuss their features.

Table A.1: Robotic systems in the reviewed literature

Type	Name (number)
Master-slave teleoperation	da Vinci (47), dVRK (6), Custom single-site (1), Other (2)
Patient-side manipulator	NeuroMaster (1), CASPAR (1), Custom orthopedic robot (2), Custom needle steering robot (7), Others (4)

Master-Slave Teleoperation-based Robot

The da Vinci Surgical System is the most popular robotic platform for AR integration, owing to its large user base. It is a master-slave teleoperation-based robotic platform. During the operation, the surgeon is seated at the console to teleoperate the robot, with help from the assistants at the bedside. One of the robotic arms holds a stereo laparoscope that captures the view inside the patient, and the stereo video is

streamed to the surgeon console, creating an immersive environment for the surgeon. By using video processing and graphical overlay, researchers are able to extend the functionality of the system. A feature of the da Vinci that is often used in AR is the TilePro^{TM} multi-input display. It allows multiple auxiliary video input channels to be displayed in a tiled manner on the surgeon console. Researchers take advantage of this feature to integrate other sources of information, for example, preoperative models [199, 200] and intraoperative imaging [166]. Since TilePro is an inherent feature of the FDA-approved da Vinci system, AR guidance implemented on it can be evaluated in clinical environments with fewer ethical and regulatory concerns. An example of a TilePro interface is shown in Fig. A.3.

The da Vinci Research Kit (dVRK) is a research platform, which reuses the robotic hardware of the da Vinci (Fig. A.4), but with custom electronics, firmware, and software maintained by an open-source community [122]. To date, the dVRK has been installed in 35 institutes around the world. As the da Vinci does not allow customization of the system due to safety and regulation, dVRK gives researchers more freedom. Among the 6 papers that use dVRK, the researchers implemented sensory substitution for haptic feedback [5, 303, 297], autonomous proton-level scanning and fusion [305, 304], and real time tracking of a user-defined safety volume [191]. Raven II [95] is another open source surgical robotic platform that serves a similar purpose as dVRK.

Laparoendoscopic single-port surgery (LESS) has emerged in recent years for la-

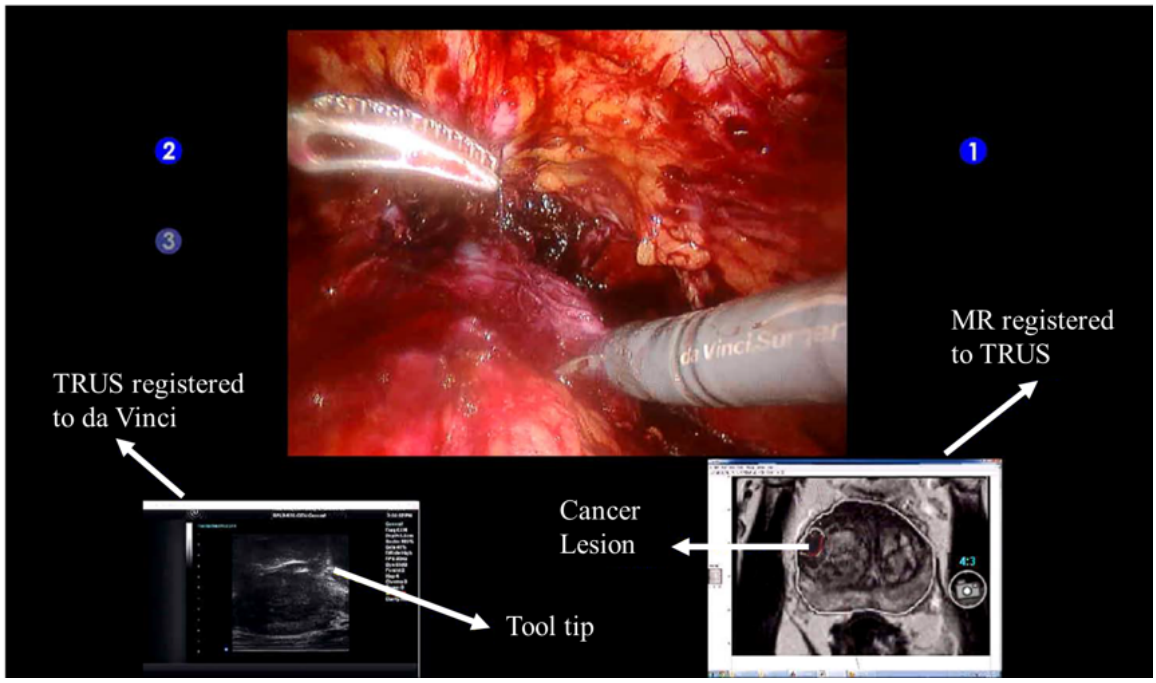


Figure A.3: With TilePro™, trans-rectal ultrasound (TRUS) and MR slices can be displayed inside the da Vinci surgeon console. It is not an AR application but AR-ready. The most relevant medical images are selected from the volume for visualization, however these augmentations are not 3D-registered with the laparoscopy. (Picture credit to Dr. Omid Mohareri)

paroscopic procedures. In most cases, it also adopts the master-slave configuration. Suzuki et al. have implemented AR-based surgical guidance with a custom LESS robot [249]. More recently, the da Vinci SP® robotic system obtained FDA clearance for urology and otolaryngology, which will open up the opportunity to integrate AR with LESS in the near future.

Patient-Side Manipulator

Many early medical robots are designed as a patient-side manipulator, e.g., NeuroMate® (Renishaw plc, Wotton-under-Edge, UK) and NeuroMaster® [41] for neurosurgery,

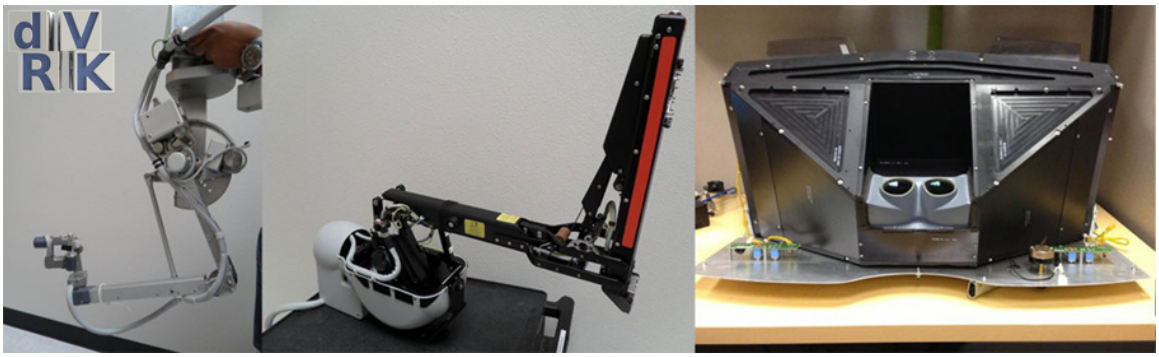


Figure A.4: Mechanical components of the da Vinci Research Kit (dVRK) are donated by the Intuitive Foundation, with community-maintained open source electronics, firmware and software.

ROBODOC® [123], Acrobot [50] and CASPAR® [292] for orthopedic surgery. 15 papers of the surveyed literature have used a patient-side manipulator. The patient-side manipulator can autonomously execute a defined surgery plan with high accuracy, but the surgeon must still be in the decision loop to ensure the operation's safety. AR can show the "intention" of the robot and allow some surgeon control. The series of work by Wen et al. is a good example of such bidirectional "communication" between the surgeon and the robotic platform [285, 281, 284, 282, 283, 280].

A.3 Application Paradigm

AR has found various use cases across different phases and types of RAS, as shown in Fig. A.2. In this section, we summarize the literature by the application paradigm.

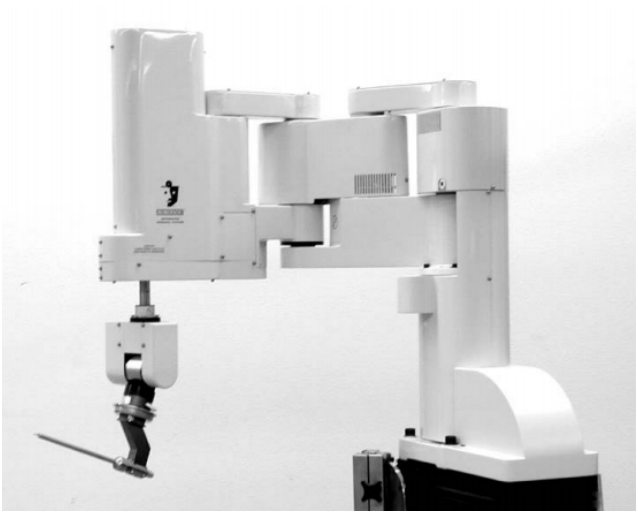


Figure A.5: ROBODOC® is an example of a patient-side manipulator for total hip and knee arthroplasty.

Surgical Guidance

AR offers the possibility to: i) highlight the critical anatomical structures or pathologies which are hidden or difficult to distinguish, ii) provide in-situ visualization of preoperative or intraoperative information about the patient or the robot, and iii) blend multiple sources of information together. Therefore, AR has the potential to be a useful aid for intraoperative guidance.

There is ample literature (47 papers) about using AR for surgical guidance in RAS. We present a diagram that summarizes various components of typical AR surgical guidance applications in Fig. A.6. The **main visual source** provides the surgeon with direct feedback of the operation. In most cases of RAS (i.e., da Vinci procedures), the main visual source is stereo laparoscopy; however, camera feed or other real-time imaging can also serve as the main visual source depending on the procedure.

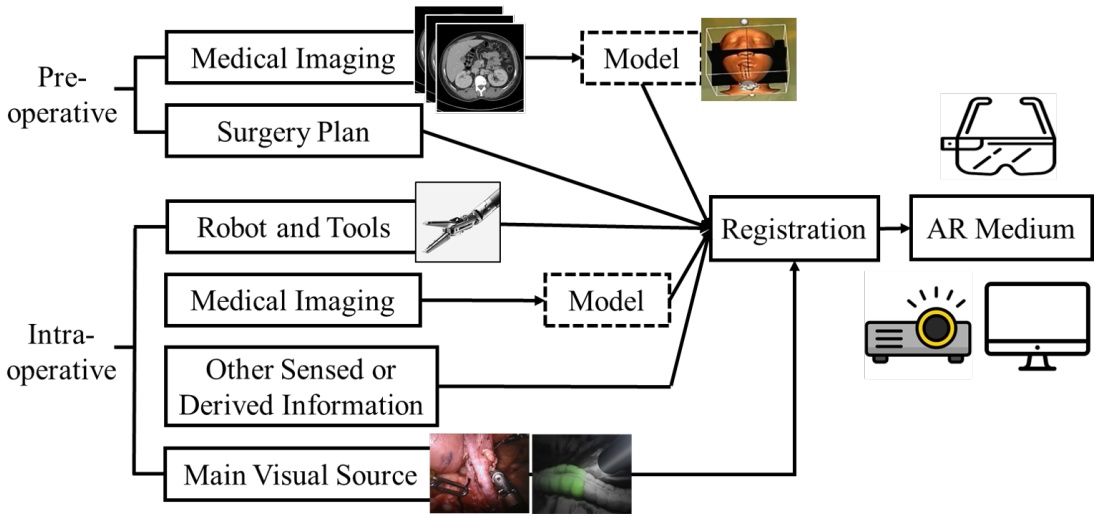


Figure A.6: Diagram illustrating the components of a typical AR-based intraoperative guidance application

All other information is visualized as graphical overlays on the main visual source. Overlays may include one or more of the following components: i) surgery plan, ii) preoperative imaging or the derived model, iii) intraoperative imaging or the derived model, iv) robot and instrument status, and v) other sensed or computed information. A **registration** process is needed to properly adjust the augmentation with respect to the main visual field. Finally, an **AR medium** presents the AR interface to the surgeon. We pick a few representative papers for detailed discussion (a full table categorizing the papers is available in [210] Tab. III).

The most “straightforward" use case of AR guidance is to bring the preoperative model into the surgeon console, displayed beside and perceptually coupled with the stereo laparoscopy. The surgeon can then view the anatomical model which clearly indicates the target structure, e.g., tumor location. As discussed earlier, this approach

is AR-ready, not full AR, because the augmented information is not overlayed on the main visual source. Volonté et al. reported the system integration and clinical case of this simple, yet effective, AR guidance system for both cholecystectomy [265, 266] and colectomy [266, 268]. A 3D mouse is integrated with the system to allow the surgeon to manipulate the preoperative model during surgery. Five patients underwent such procedures and the authors claimed that AR offered "undeniable help" during the dissection phase [266].

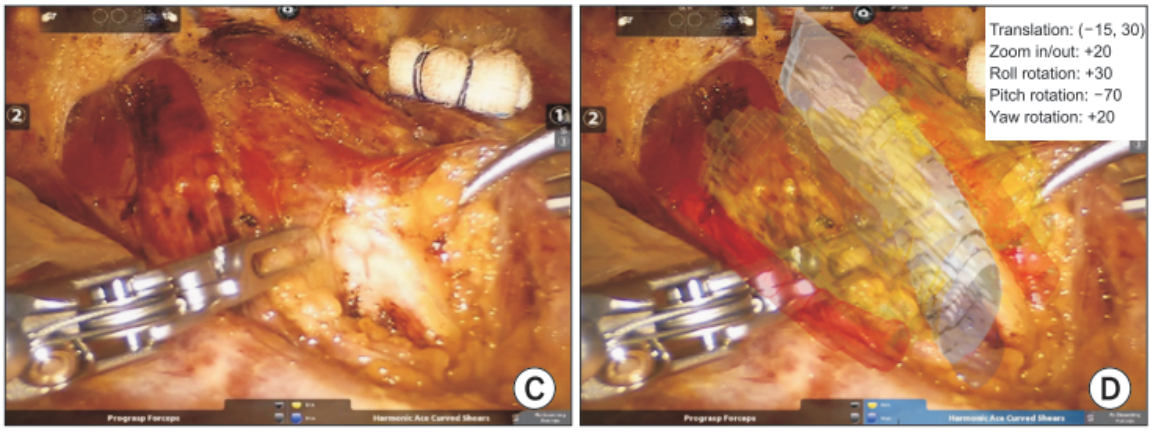


Figure A.7: AR for surgical guidance with preoperative model. The anatomical model (thyroid gland, common carotid arteries, trachea, and esophagus) is manually aligned with the laparoscopic image during robotic tracheal exposure of thyroidectomy (Fig. 3c and 3d of Lee et al. [138]).

One step beyond the "straightforward" approach is to register the model with the laparoscopy, via manual alignment [138] (as shown in Fig. A.7), fiducial-based registration [146, 147], optical-flow-based semi-autonomous registration [227] or other advanced algorithms [203, 96]. Liu et al. used fiducials implanted in the target model to register the preoperative model to the laparoscopy and the setup was evaluated in otology surgery [146] and transoral surgery [147].

Coste-Manière et al. investigated AR guidance for robot-assisted cardiac surgery, specifically, totally laparoscopic coronary artery bypass (TECAB) [46]. In this series of work [173, 46, 62], they proposed to overlay preoperative models (CT scan of the heart, coronary angiography) on the laparoscopic image. The registration is performed by either using the robotic instrument to point at pre-installed fiducial markers [173, 62], or by using a projector-camera system to register to the outer surface of the body [46]. Robot kinematics data is used to maintain the alignment. Although the preliminary accuracy of the overlay is insufficient (error ranges from 9.3 mm to 19.2 mm) due to anatomical deformation, the authors felt that the experiment results were rewarding. Voruganti et al. took a similar approach and the pre-clinical results showed that the accuracy was also insufficient [270]. The highly deformable nature of the cardiac area and the coronary tree has been a significant challenge for achieving accurate AR overlay. Figl et al. proposed to reconstruct a 4D heart model for registration that considers the phase of the cardiac cycle [68].

Apart from showing the preoperative model, AR can also blend intraoperative imaging or models with the surgical scene. Leven et al. [141] and later Schneider et al. [226] proposed "flashlight" visualization to overlay the intraoperative ultrasound image onto a 3D representation of the imaging plane in the stereo view of the console, as shown in Fig. A.8. Adebar et al. and Mohareri et al. integrated robotic Transrectal Ultrasound (TRUS) with the da Vinci for radical prostatectomy [3, 167, 165]. Adebar et al. developed a drop-in tool for registering TRUS images with laparoscopic

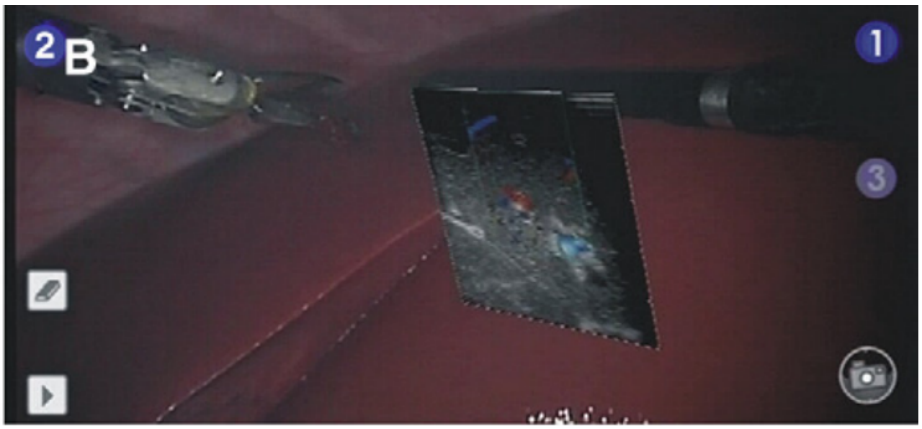


Figure A.8: AR for surgical guidance with intraoperative imaging. Laparoscopic ultrasound image is overlaid on the probe using the proposed "flashlight" visualization, displayed in the surgeon console (Fig. 2b of Schneider et al. [226]).

video [3]. Mohareri et al. took advantage of the registration to control the TRUS probe to track the tip of the laparoscopic instrument. Therefore, the TRUS sagittal image plane can always follow the surgeon's manipulation and provide real-time ultrasound imaging of the anatomical region of interest [167, 165]. In the aforementioned papers, the intraoperative medical images are not visualized on top of the surgery scene, therefore they present AR-ready scenarios according to our definition. Fuerst et al. developed the first robotic SPECT for sentinel lymph node mapping, where the reconstructed SPECT volume is fused with the laparoscopic video [71]. The SPECT probe is small enough to be inserted into the surgery site and grasped by a robotic instrument. Gorpas et al. integrated autofluorescence lifetime of the tissue as augmentation for surgical guidance [80].

A novel and interesting approach, proposed and developed by Edgcumbe et al., uses a projector-based AR intracorporeal system (PARIS) [59]. A miniature projec-

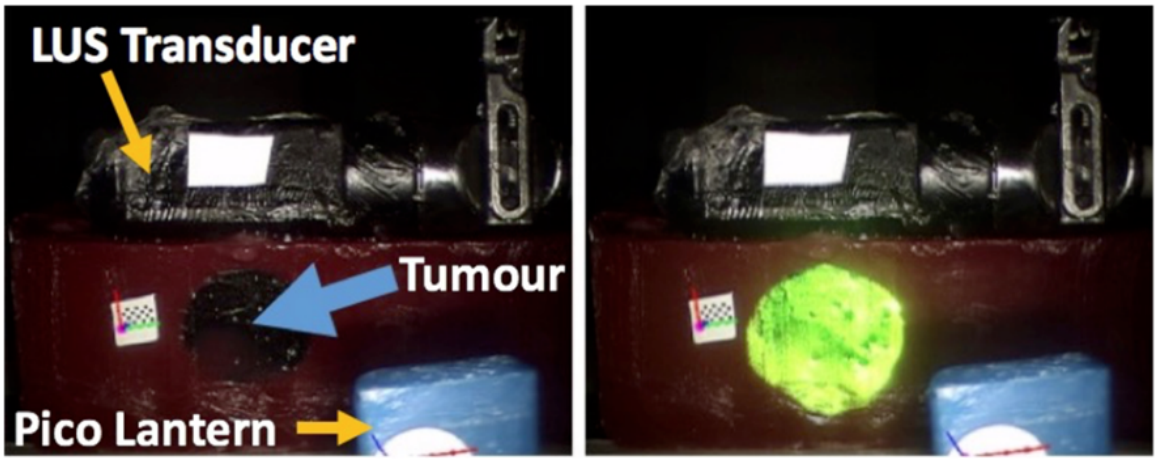


Figure A.9: Intracorporeal AR: tumor margin (green color) projected on phantom liver surface using Pico Lantern (Part of Fig. 4 of Edgcumbe et al. [59])

tor, called Pico Lantern, can be dropped into the patient body and picked up by a laparoscopic instrument in a da Vinci surgery [57]. After calibration and registration, the Pico Lantern can project the tumor margin directly onto the organ surface with an RMS error of 0.8 mm, as shown in Fig. A.9.

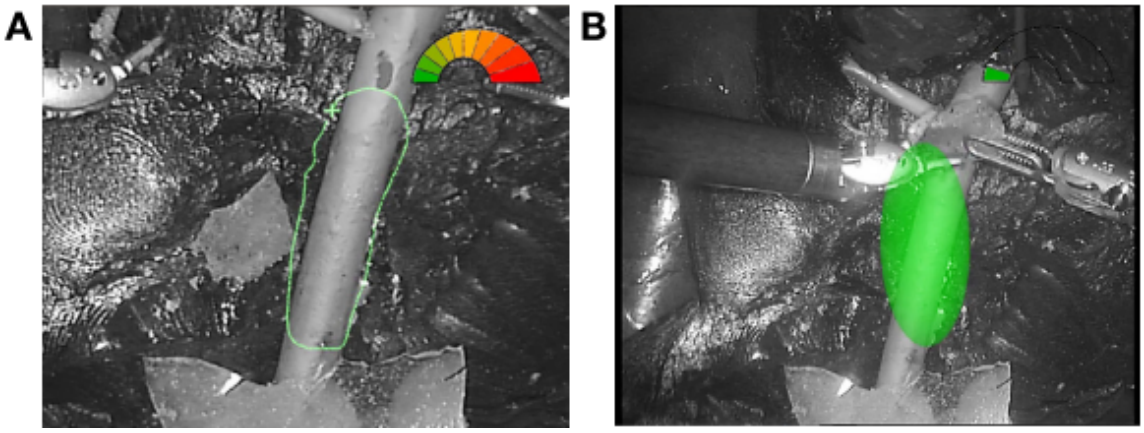


Figure A.10: AR for surgical guidance with derived information in the surgical field: a user-defined safety volume, and the distance indicator between the surgical instrument and the delicate area (Fig. 8 of Penza et al. [191])

With intelligent algorithms, it is possible to compute additional information from

the surgical site and therefore provide the surgeon with additional guidance information. Penza et al. developed EnViSoRS, which is able to use a real time stereo reconstruction algorithm to track a user-defined safety volume during robotic surgery [191]. As demonstrated in Fig. A.10, AR visualization is used to inform the surgeon about the current distance between the instrument and the delicate area.

Interactive Surgery Planning

Researchers proposed to use AR interfaces to help surgeons create surgery plans for tumor ablation procedures [285, 281, 283, 282, 284, 280], vocal fold microsurgery [280], stereoelectroencephalography (SEEG) implantation [302], tele-neurosurgery [41, 162], and robotic prostate biopsy [79]. Spatial AR has been mostly applied for interactive surgery planning, offering the advantage of i) visualization overlaid on the patient body, and ii) hand-gesture based interaction for adjusting the surgery plan. Another potential advantage is that hand gestures allow the surgery team to remain sterilized, unlike many other touch-based interfaces.

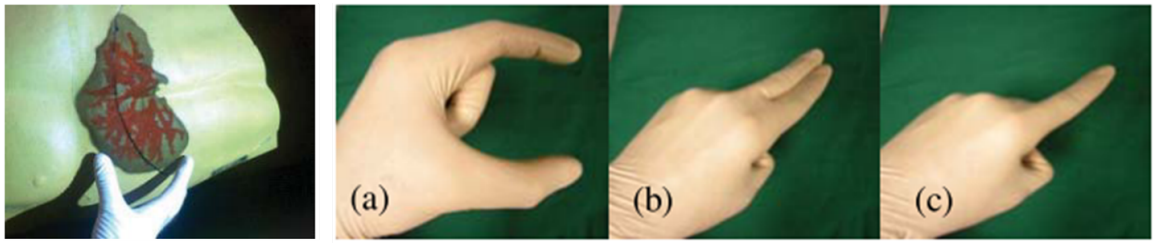


Figure A.11: Projector-Camera AR system that allows hand gestures interaction for surgery planning (Fig. 7 and 8 of Wen et al. [285])

Wen et al. applied the projector-based AR system with a custom needle-steering

robot in a series of publications [285, 281, 283, 282, 284]. The surgeon can view the preoperative model and the ablation plan projected onto the patient body and interactively adjust the plan with hand gestures [285, 283, 284] (Fig. A.11) or via the computer workstation [281]. In [280], the author extended the previous work to use a tablet rather than a projector for a video see-through AR system. The touch screen of the tablet is used for interaction with the surgery plan. The setup is also tested in vocal fold microsurgery.

Chou et al. proposed to use an AR interface to create, preview and interact with the surgical plan in tele-neurosurgery [41]. The surgeon can simulate and verify the surgery plan before it is transferred to the remote site where a NeuroMaster robot will execute the plan. The approach was tested in several clinical cases [162].

In traditional robot-assisted SEEG, the surgeon can only observe the plan from a monitor. Zeng et al. proposed to use a projector-camera system to provide in-situ visualization of the SEEG implantation, directly overlaid on the patient's head, thereby allowing the surgeon to verify the accuracy of the implantation [302]. The projection error is evaluated to be $0.82 \pm 0.23mm$. Gîrbacia et al. adapted a similar idea to plan the trajectory of a biopsy gun [79].

Port Placement

Port placement is an essential step before robotic-assisted laparoscopic surgery [66]. The laparoscope and the robotic instruments are inserted through planned ports.

Good port placement can avoid potential collision of instrument shafts, allowing maximum access and visualization of critical areas [277]. Similar to interactive surgery planning, the port placement plan can be visualized using Spatial AR technologies.

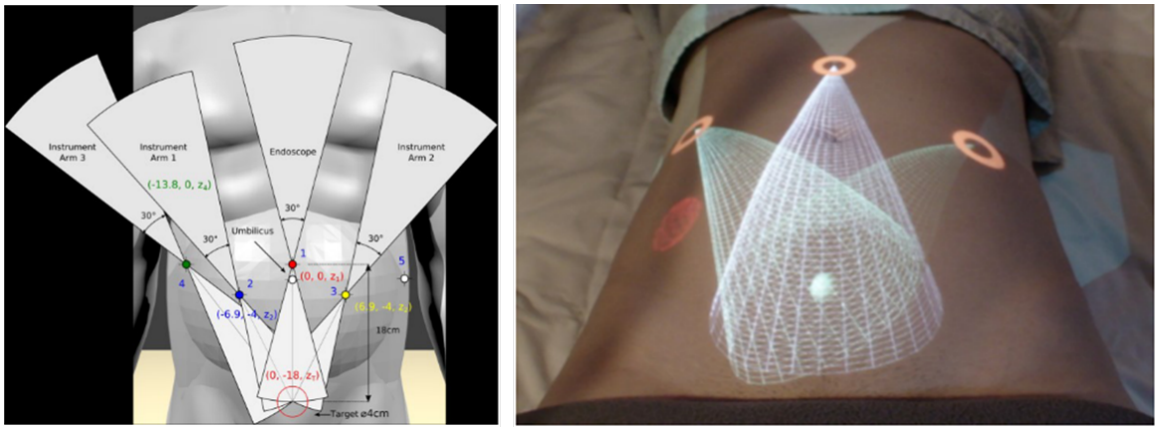


Figure A.12: Leonardo: projector-camera-based port placement planning (Fig. 3 and 7 of Simones et al. [232])

Coste-Manière et al. first proposed to use AR to facilitate port placement [46, 62]. It is hypothesized that the surgery team can identify potential collisions more easily, with virtual instruments overlaid on the patient anatomical model (skin, ribs and target anatomy). Apart from projecting the ports and instruments, Wörn et al. proposed to calculate a goodness value and project a heatmap of it onto the patient's body [293]. Weede et al. studied the optimization problem more systematically, by considering the reachability of the target areas, collision avoidance, dexterity and ergonomic factors [277, 278]. Simones et al. presented Leonardo, which used a gesture-based interaction framework to plan port locations [232] (Fig. A.12).

Advanced Visualization of Anatomy

AR is essentially a set of advanced data presentation methods. In some cases, the researchers simply present the AR interface to the surgery team without a specific clinical aim like planning or surgery guidance. We categorize these works under the paradigm of advanced visualization of anatomy.

Kolagunda et al. proposed to use Oculus or HTC Vive to visualize the 3D model of the prostate, bladder and tumor aligned with the model reconstructed from stereo laparoscopic images for radical prostatectomy [129]. The hypothesis is that overlaid visualization can improve spatial awareness and therefore help decision making at key stages of the procedure.

Huber et al. experimented with offering surgeons, assistants and trainees a “virtual monitor" through HoloLens in robotic-assisted Transanal Total Mesorectal Excision (taTME) [108]. Each surgery team member can place their own “virtual monitor" at the most comfortable location, without interfering with other staff. In a traditional setup, the locations of external monitors need to be carefully selected to cater to each OR staff. The HMD has the potential to be a more convenient visualization platform than an external monitor.

Supervised Robot Motion

Sometimes the surgical robot is designed to autonomously perform specific task, especially for the patient-side manipulator-type robot. During the period of autonomous operation, it is critical for the surgeon to be able to closely monitor the robot's motion, or the "intention" (i.e., the action plan), in order to ensure safety when the robot malfunctions. AR can provide the interface for in-situ visualization of the motion and intention of the robot.

In the series of work by Wen et al. [285, 281, 283, 282, 284, 280], the projector-based or tablet-based AR system provides intraoperative visualization of the trajectory of the needle held by the robot, together with the preoperative model and the ablation plan. The surgeon can check whether the robot path is valid. The accuracy of the system is measured to be between 1.74 mm and 2.96 mm in a phantom study [280]. Moreover, the surgeon is able to use hand gestures to directly control the motion of the robot [284].

Sensory Substitution

Although it is desirable for the surgical robot to provide haptic feedback to the surgeon directly during manipulation, achieving it remains a technical challenge. Sensing the force and rendering it as a visual feedback provides an alternative solution to direct haptic feedback. In this case, AR techniques can be used to integrate the visual

haptic feedback in the conventional visualization interface.

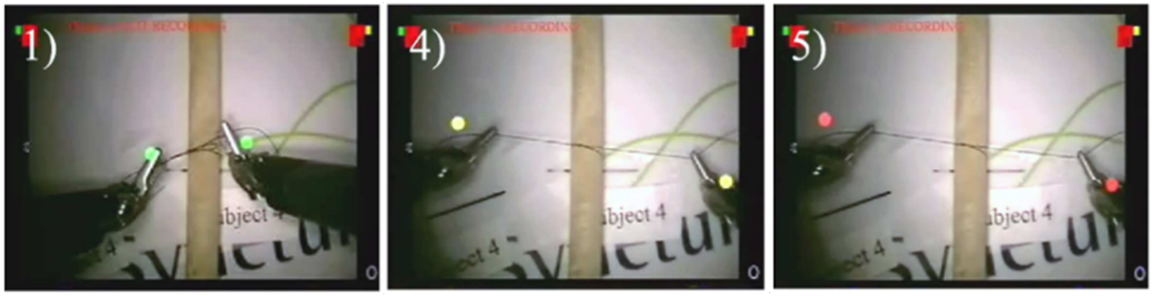


Figure A.13: Force feedback on surgical instrument is substituted as graphical overlay for the surgeon during knot typing. The color of the nearby circle represents the extent of the force being applied. (Fig. 1 (1,4,5) of Akinbiyi et al. [5])

Akinbiyi et al. integrated a force sensor on a da Vinci instrument and proposed to visualize haptic feedback using AR on the console in da Vinci surgery [5]. The authors categorized the extent of force into low, ideal, and excessive force zones. Then, they render a sphere, colored depending on the current force category, as shown in Fig. A.13. The location of the rendered sphere is registered with the location of the tool tip, therefore, the surgeon can be aware of the force being applied on the instruments during the operation. A multi-user study with one surgeon and eight non-surgeons showed that the number of broken sutures and the number of loose knots were both decreased with AR-based sensory substitution.

Yamamoto et al. developed an autonomous tissue stiffness estimation system for robotic-assisted surgery [297]. The robotic arm is able to autonomously palpate the anatomy, and the computed stiffness value is displayed on the tissue using the hue channel. In later work, the authors extended this method to display a 3D stiffness map overlaid on the reconstructed mesh of the anatomy [296]. Instead of relying on

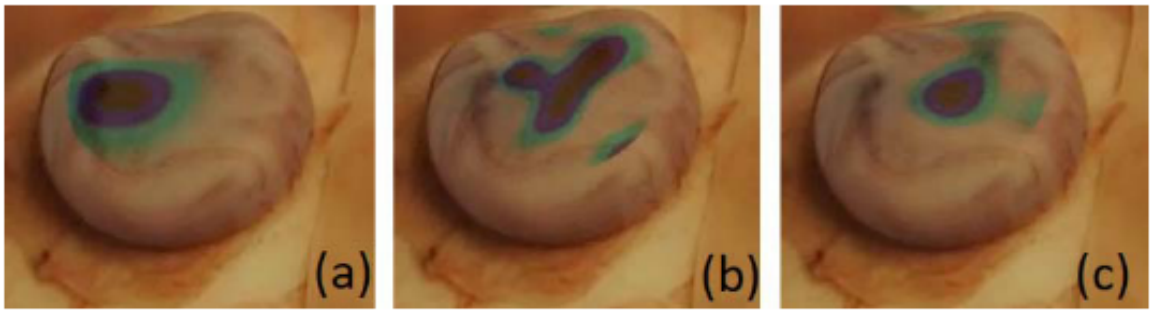


Figure A.14: Stiffness property of the tissue is rendered as 3D AR overlay (Fig. 5 of Zevallos et al. [303]).

the reconstructed 3D mesh of the tissue, Zevallos et al. first registered the laparoscopy with a preoperative model of the tissue and then projected the stiffness map onto the 3D model [303], as shown in Fig. A.14.

Bedside Assistance

The bedside assistant plays an important role in robotic-assisted surgery [229]. AR can benefit the perception [24] or the performance [207] of the bedside assistant.

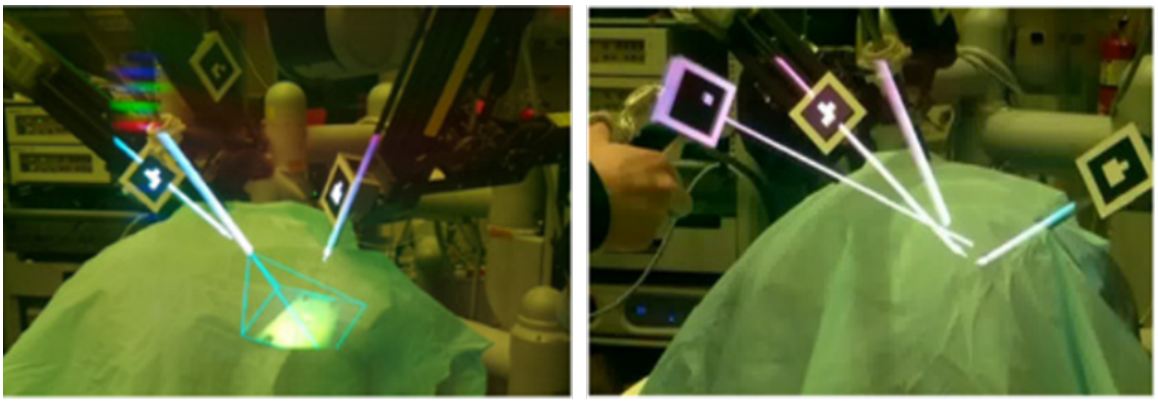


Figure A.15: The see-through view with ARssist (Fig. 4 of [207]). The bedside assistant is able to see the "hidden" robotic instruments, hand-held instrument, and laparoscope through a HoloLens.

Qian et al. proposed ARssist to aid the patient-side assistant of da Vinci surgery, with an optical see-through HMD [207] (Fig. A.15). The details of ARssist is presented in Ch. 5.

Skill Training

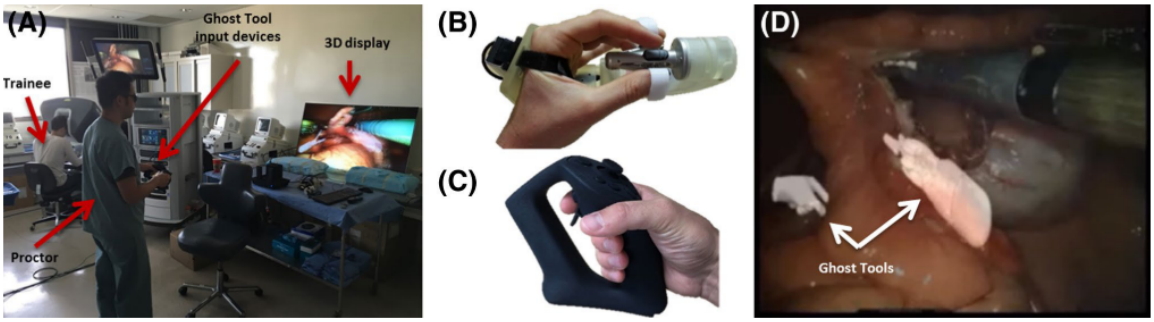


Figure A.16: AR used in proctor-trainee-based surgical procedural training [116]

Jarc et al. employed AR techniques in proctor-trainee training scenarios for RAS, where the proctor is able to control the ghost tools that are rendered as augmentations on the trainee's immersive display [116]. The preliminary evaluation with a limited number of participants found that both proctor and trainee favored the AR-based mentoring technique. Matu et al. reported a similar proctor-trainee scenario, but a HMD was used for augmented visualization instead of a 3D monitor [160].

A.4 Clinical Relevance

The ultimate objective of a new surgical technology is that it brings clinical advantages, e.g., improving surgical outcomes, efficiency or affordability. Combining AR with RAS's clinical contribution requires careful evaluation. We summarize the evaluation methods among the surveyed literature in Tab. A.2.

Table A.2: Evaluation methods in the literature

Type	No.	Type	No.
Simulation in silico	5	Cadaver ex vivo	2
Phantom in simulacra	21	Animal in vivo	13
Phantom ex vivo	11	Human in vivo	19

Although simulation studies and phantom studies provide valuable feedback about certain methods, their clinical translation is hard to predict. Therefore, we highlight the papers that are evaluated in a clinical setting (cadaver ex vivo, animal and human in vivo), and discuss their findings and special considerations, both positive and negative, for adopting AR in RAS. In aggregate, there are 274 subjects (6 cadavers for ex vivo studies, 25 animals and 243 patients for in vivo studies).

Proof-of-Concept

As AR is a premature technology for surgery, the aim of many publications included in the survey is to demonstrate whether using AR in specific RAS procedures is

feasible. The concept of AR-integrated RAS is shown to be feasible in partial nephrectomy [202], cholecystectomy [100], TECAB [62], radiofrequency ablation [281], radical prostatectomy [166, 129, 199, 200], mandibular angle split osteotomy [307], thyroidectomy [138], lung segmentectomy [20], cochlear implant procedure [146], and transoral surgery [147]. No complication has been reported in the surveyed literature.

Reduced Sight Diversion

Sight diversion is a common issue in image-guided surgery, where surgeons need to pay attention to various sources of information in the operating room. More specifically, in image-guided RAS, surgeons need to switch their focus between the laparoscopic video, the other medical imaging monitor, and the patient body. Displaying information in one place, therefore reducing sight diversion, is an obvious advantage of AR (or AR-ready) systems. Researchers have identified this potential advantage and evaluated the clinical benefit.

Volonté et al. reported a case study of AR-ready totally-robotic right colectomy [268] using TilePro. During the procedure, the surgeon switched to the co-located AR-ready visualization 5 times. Similarly, Mohareri et al. measured the time that the surgeon turns on the TilePro view with MRI slices and the real time TRUS image to be 16 and 28 minutes in two clinical cases [166] (Fig. A.3 shows an example view through TilePro). It is not a trivial task to quantify the benefit of reduced sight diversion, which would involve statistical comparison of operations in identical

clinical conditions with and without the AR (or AR-ready) assistance. In the above papers, the authors presented the clinical case study with the option of AR assistance and hypothesized that the surgeon would have otherwise spent more time without AR assistance simply due to the physical inconvenience of switching focuses.

In contrast to [268, 166], Huber et al. deployed AR for robotic-assisted taTME, a two-team collaborative clinical procedure, to study the effect of reduced sight diversion in a cluttered clinical environment [108]. In robotic taTME, two surgery teams, the robotic team and the transanal endoscopic team, collaborate in the OR. Multiple sources of information are displayed in the separate monitors positioned in different locations of the OR. Without AR, monitor positioning is a challenge because each person on both surgical teams needs a clear and comfortable view of both videos. With the HoloLens, the “virtual monitors" can be positioned at the most comfortable locations in the virtual space for each person, while not interfering with other team members. According to Huber et al., the proposed mixed reality setup reduced sight diversion for all members of the operating team (surgeon, assistant and trainee). The subjective evaluations in [108] revealed a high comfort level despite the heavy weight of the HoloLens.

The literature demonstrates the preliminary benefit of reduced sight diversion in image-guided RAS. The fusion of multiple imaging sources decreases the need to change the viewer’s focus in the OR. For a collaborative setup, the reduced sight diversion via AR “virtual monitors" can increase the comfort level of different team

members.

Improved Situation Awareness

Situation awareness is of critical importance for the surgeon to make interventional decisions. In some cases of RAS, the current situation is not easily observable from the laparoscopic video, especially when the target lesion is occluded or not visually distinguishable. By visualizing additional information of target anatomy or lesion via AR, ideally registered with the laparoscopic video, AR-based surgery guidance can improve the surgeon's situation awareness.

In a clinical case of radical prostatectomy, presented by Mohareri et al. [166], the surgeon's ability to see a lesion on the left posterior side in MR images via augmented reality changed the surgeon's clinical decision to not do nerve-sparing on that side. Gorpas et al. also target tumor removal, but use autofluoresence to differentiate types of tissue to guide the surgeons to obtain a positive margin [80]. Similarly, Liu et al. [147] color-code and label depth and distance to the ideal margin and critical structures in arterial dissection and base-of-tongue neoplasm resection. Tumor resection on ex vivo animal models failed for procedures with fluoroscopic guidance, but succeeded in all cases with AR guidance. Zhou et al. use HMD-based AR in mandibular angle split osteotomy to visualize the inferior alveolar nerve to avoid neurosensory disturbance [307]. Lee et al. uses monitor-based AR in thyroidectomy to replace the tactile feedback surgeons would feel from probing different structures

in traditional laparoscopy [138]. Porpiglia et al. conducted a clinical study of robot-assisted radical prostatectomy with 40 patients (20 in traditional setup, 20 with AR visualization of deformable preoperative model) [200]. In this study, it was reported that the AR visualization led to marked improvement of identification of capsular involvement.

In all the above clinical studies, AR provides additional information to the surgeon which aids in making clinical decisions, such as providing additional information about the tissue [80, 138] or showing surrounding structures [166, 307, 200], and potentially can extend RAS to novel fields [147]. These additional source of information provide critical knowledge about the current clinical situation to the surgeon.

Accuracy Challenges

When AR is used to superimpose virtual 3D objects at a desired position and orientation (registered with the anatomy), accuracy of the overlay is critical to prevent the surgeon from making a wrong decision due to the “misleading" situation awareness. For instance, a poorly registered AR system may display the tumor at a wrong location, then if surgeons make surgical decision purely relying on AR, they could damage healthy tissue while leaving the tumor intact.

Many different registration methods have been proposed (autonomous, semi-autonomous or manual), and some of them have been evaluated for surgery guidance in RAS. Autonomous or semi-autonomous (involving manual initialization) registration methods

are appealing in terms of the ease of setup. For cardiac procedures (TECAB), Falk et al. reported an overlay accuracy between 9.3 mm to 19.2 mm using fiducial markers and robot kinematics [62]. The large error is due to heart deformation. Liu et al.'s method [147] used fiducial markers and manual initialization, achieving 2 mm error in registration but still had 5 mm mean tool tracking error. Mohareri et al. developed two methods to register a TRUS and da Vinci robot for AR guidance, and achieved target registration error (TRE) of around 2-3 mm [167]. As we can see from the reported numbers, the ideal sub-millimeter accuracy is not yet achieved.

Many other researchers use manual registration to make sure the preoperative model appears correctly overlaid on the laparoscopic video in clinical studies, e.g. [138, 199, 200]. In this case, the surgeon, or an assistant that is knowledgeable about the clinical situation, is responsible for the alignment. Such solution presents a trade-off between the efficiency and the reliability of the AR system.

The requirement for overlay accuracy is dependent on the specific procedure and the type of clinical information to be displayed. A suitable registration method should be chosen after the accuracy requirement is defined. However, there is a lack of literature that properly defines the clinical requirement for AR-integrated RAS and reviews possible technical solutions of registration for each level of accuracy requirement.

Additional Setup Time

Extra setup procedure is required for AR-integrated RAS to function in the operating room. The additional setup may include hardware installation [267, 166], planting fiducials on the patient or on the hardware for tracking [147, 144], software computation for registration [62], and user interaction to adjust graphics (position, color and transparency). In the case where an intraoperative model is used for AR overlay, it requires additional time to obtain and compute the intraoperative model [80].

More specifically, Falk et al. reported 3-8 minutes for calibration of the AR system for robotic-assisted TECAB [62]. Volonté et al. reported 10 minutes as the projector setup time to overlay the anatomical model on the patient abdomen [267]. Mohareri et al. reported that the setup of a TRUS system for AR requires on average 7 minutes, and the registration between TRUS and da Vinci costs less than 2 minutes [166]. Zhou et al. used a special bone-mounted marker for AR surgical guidance, which requires considerable amount of time for the surgeon to drill 3 holes on the mandible bone to affix the marker [144]. Gorpas et al.'s method to use a laser to map the tissue takes 5 min [80].

According to a study by Childers et al., the average cost of OR time is $36 to $37 per minute [38], not including the extra cost for robot operation. Therefore, the additional setup time will add further cost to the already expensive RAS. The cost efficiency of AR-integrated RAS needs to be carefully evaluated with respect to the clinical benefits.

Activation on Demand

Activation-on-demand is the technique by which the surgeon can activate the AR interface at key stages of the surgery to confirm a surgical decision, but the traditional view is shown the rest of the time. We identified that activation-on-demand is a commonly implemented feature in the literature of AR-integrated RAS, especially when such application is used in a clinical setting. In [202, 100, 268, 167, 165, 166], the surgeon uses a foot pedal to activate the enhanced AR visualization on the surgeon console. In [129], the surgeon steps out of the console and observes the immersive AR visualization with a VR headset. Although it is cumbersome to switch the visualization, the surgeon still typically activates it prior to bladder neck sparing and apical dissection. Activation-on-demand has a variety of advantages. First, it minimizes the disturbance to the current workflow because AR provides navigational guidance only at critical moments and when the surgeon demands it. Secondly, the surgeon does not make surgical decisions based purely on AR because the normal visualization was presented to the surgeon right before he or she activates AR visualization, which reduces the risk caused by the bad AR alignment.

Visual Clutter of AR Interface

Visual clutter refers to the situation where the user interface is overwhelmed with excess or disorganized information, and causes decreased recognition perfor-

mance [221]. In AR-based surgical guidance, if the overlay information is not well filtered or organized, it potentially causes visual clutter for the surgeon, imposing the risk of bad surgery decisions. Visualization techniques are required to reduce the visual clutter and the subsequent perceptual load. [100] and [20] use transparency of the augmented model to reduce visual clutter. Volonté et al. allows the surgeon to manipulate the window level of the projected image on the patient body, to navigate from the patient skin to the bone [267]. Activation-on-demand is also one solution that addresses the issue in the temporal domain, so augmentations only appear at the times when they are needed.

Visualizing the Occluded Anatomy

Simple rendering, e.g., making the overlaid image semi-transparent, is applied in the majority of the surveyed literature. Fig. A.7 demonstrated one such technique for thyroidectomy. It is useful to reduce visual clutter. However, if the anatomy is supposed to be occluded behind the tissue, but rendered as a semi-transparent graphical overlay, the surgeon may suffer from a loss of depth perception [140]. Lerotic et al. developed a novel pq-space based non-photorealistic rendering technique for robotic lung lobectomy. A retrospective evaluation of this method showed that the error in perceived depth is significantly smaller [140].

In the medical AR domain, researchers have proposed more advanced methods which consider the surface property, the viewing perspective, and instrument location

to improve depth perception and enable motion parallax for in-situ visualization of anatomical structures [23]. Such techniques will potentially benefit many application scenarios of AR-integrated RAS, e.g., surgery guidance, planning and bedside assistance. Although no complication is reported with the simple AR scheme, advanced AR visualization methods that provide better depth perception will encourage clinical adoption.

From AR-Ready to AR

As described in Sect. A.1, we identified 8 AR-ready systems, where the augmentation is correctly oriented [266, 268, 265, 20, 201] or selected from the volume [167, 165, 166], but is not overlaid on the main visual source, as required by the definition of Azuma et al. [16]. In fact, some of the above systems have been tested in patient studies with relatively large sample size (20 patients in [165], 45 patients in [20] and 52 patients in [201]). Creating the “overlay" is the last step for such systems to become AR. However, there exist many technical challenges to overlay the separate visual streams comfortably, to deliver information efficiently, and allow intuitive interactions, and to be robust to the dynamic clinical environments. With these concerns, AR-ready systems are currently easily accepted for clinical use, as they are simpler to implement (e.g., split-screen display) and less risky.

Fail-Safe of AR System

Safety is always the top priority in surgery. As AR is still relatively untested technology in surgery, consideration must be given that it "fails safely". Activation-on-demand helps to mitigate the issue to some extent, because the surgeon does not make surgery decisions purely based on the augmented view. In case of AR system failure, an experienced surgeon will be able to perceive the conflict, having the opportunity to identify the unexpected behavior. An OST-HMD-based AR system is also fail-safe [218]. For instance, in the case study of Huber et al., the regular high-definition monitors were always available in the background, so that the procedure could safely continue when some error occurred to the "virtual monitors" on the HoloLens [108].

Additional Roles for the Assistant

Surgeries today are team work, where the assistant plays an important role. In AR-integrated RAS, the assistant needs to help with tasks that are specific to the AR interface, for example, to manually adjust the graphical overlay [202, 199, 200, 138]. In the clinical setting, the overlay generated by an experienced assistant may still be considered more accurate than computer registration. Apart from the accuracy, by offloading the alignment task to the assistant, the augmentation can be prepared in parallel during the surgeon's operation, making the process more efficient. In [227], the assistant refines the overlay after the initial automated registration.

Sterilization

While sterilization is not of concern in most teleoperation setups, as the master console is generally not sterile, it is of concern when the surgeon is at the patient's side. This limits the available interfaces. Baste et al. uses Therapixel^{TM} (Valbone, France) to visualize the preoperative lung model, which allows hand gesture manipulation [20]. This feature enables the surgeon to remain sterile during the procedure. Sterilization is also of concern when designing new equipment. Gorpas et al. used sterilizable fibers or a sterile sheath over the fiberoptic probe [80]. It is also important to note that the surgeon's head is not sterile and therefore a surgeon cannot use his or her sterile hands to adjust a HMD, unless a sterile handle is attached to the HMD.

A.5 Future Perspectives

Expansion of the Application Paradigms

AR-integrated RAS has been applied in various application domains and clinical procedures, as demonstrated in Sec. A.3. The most frequent application paradigm so far is surgical guidance, where models are often manually constructed. Only recently have researchers started to evaluate AR-integrated RAS for bedside assistance and patient education. It is expected that the application domains will keep expanding.

Improved Hardware Platforms

The hardware technologies for AR and RAS are both rapidly evolving. In terms of RAS platforms, Intuitive has announced the 510 (k) clearance of Ion, a robotic endoluminal platform for minimally invasive peripheral lung biopsy. Auris Health Inc. (Redwood City, CA) obtained clearance for its Monarch® surgical robot which targets peripheral bronchoscopy. With the flexible endoscope technology in these commercial systems, we can foresee advantages of improved situation awareness brought by AR at the bedside. Meanwhile, MagicLeap (Plantation, FL), Microsoft (Redmond, WA) and Oculus (Menlo Park, CA) are pushing the capability of the HMD forward. Cost-efficient and convenient devices encourage early experimental use in the operation theatre.

Specialized Hardware Platforms

AR platforms and surgical robots are mostly treated as separate modules in the current literature. They require separate setup, e.g., registration and data communication. The separation of both systems may limit the stability and usability of AR-integrated RAS, for example, Huber et al. reported the instability of the connection between HoloLens and the server for endoscopic video streaming. In the future, specialized hardware may appear for AR-integrated RAS, which can minimize the overhead for setup, support high bandwidth of data communication, preserve a

smooth clinical workflow, and cater to the specific clinical requirements.

Clinical Evaluations

We believe various clinical evaluations of AR-integrated RAS are in progress and more will appear in the future. Current evaluations are mainly exploratory feasibility studies, which are an indispensable first step for the community. However, there is little evidence about whether there is actual improvement in terms of clinical outcome, nor statistical significance of the improved surgeon comfort. Clinical evaluations with more samples and specific clinical objectives is a future perspective.

Reliable Software

Powerful software can ease the setup for AR-integrated RAS and improve user experience. For example, while manual registration has been used in existing works [246], autonomous registration would reduce the burden on the surgical team. As discussed in Sect. A.4, the alignment task is mainly performed semi-autonomously or manually, for the simple reason that the advanced methods are not yet reliable enough by clinical standards. Other examples include surgical scene reconstruction [211] and segmentation [183], autonomous tumor localization [296], heart beat stabilization [243], safety volume tracking [191], computer-aided diagnosis [294] and advanced rendering [140]. There is still a significant gap before the proposed algorithms are reliable enough for

clinical deployment.

Deep Learning

Some works have started to bring the achievements of deep learning in perception to robotics and explored how to use AR to display them. Wang et al. show that augmenting depth to monocular endoscopy leads to better task completion in a phantom [275]. Liu et al. explore reconstructing depth images in monocular endoscopy using self-supervised learning [148], which can be used in real patients with the previous augmentation. Although these works do not explicitly address robotic systems, it can easily be extended for such.

The community is leading the technology transition. MICCAI (International Conference on Medical Image Computing and Computer Assisted Intervention) hosts an annual challenge for surgical scene understanding - from binary segmentation of instrument vs. background, to intra and interclass tool segmentation and labeling patient anatomy. The 2018 winning entries both incorporate a similar base module that has been pretrained on ImageNet [54]. This shows the potential to bring the success of neural networks in natural image understanding to the medical domain, which can possibly contribute to AR-integrated RAS in the future.

A.6 Conclusion

We reviewed the literature that applied AR to RAS. The papers are classified by their application paradigms, which include surgical guidance, interactive surgery planning, port placement, sensory substitution, supervision of robot motion, advanced visualization, bedside assistance, and surgery skill training. The majority of the papers fall into the paradigm of surgical guidance, which aims to use AR to provide inaccessible or hard-to-access information for the surgery team intraoperatively. We discuss the hardware components of the robotic system and AR medium that are applied in the literature, each with its own advantages and disadvantages. In the future, integrated hardware platforms with improved stability and accuracy may appear, which target specific clinical procedures.

As an emerging surgery technique, AR-integrated RAS has been experimentally deployed for clinical evaluations. Preliminary studies have demonstrated the feasibility of such an approach, and identified the benefit of reduced sight diversion and improved situation awareness for the surgery team. Throughout the trials, clinicians and engineers have gathered valuable experiences and insights. For instance, activation-on-demand is a commonly applied technique to ensure smooth clinical workflow and a gradual adoption of AR interfaces.

Appendix B

How to Compile *ARssist*

This appendix details the compilation procedure of *ARssist*, an AR application on Microsoft HoloLens v1 implemented with Unity. The methods and evaluations of *ARssist* are presented in Ch. 5. The developer is assumed to have good understanding of computer science, programming experience with Windows and Linux, and development experience with Unity and HoloLens. The following instructions are for using *ARssist* with a da Vinci Si robot via its research interface. Other possible configurations include using *ARssist* with the da Vinci Research Kit (dVRK).

1 Requirements

First, the developer needs to access the repositories on the LCSR internal git server (https://git.lcsr.jhu.edu/aramis), including:

1. FFmpegUnityInterop: https://git.lcsr.jhu.edu/aramis/ffmpegunityinterop
2. HoloLensARssistSi: https://git.lcsr.jhu.edu/aramis/hololensarssistsi
3. DeckLinkSteam: https://git.lcsr.jhu.edu/aramis/decklinkstream

1.1 Requirements and Setup for Compilation

The developer needs the following hardware and software (the version information is the one I am using, other versions not guaranteed to work):

1. Microsoft HoloLens v1
    - With developing mode, not necessarily research mode
    - OS version: 10.0.10240.0 to 10.0.17763.0
2. da Vinci Si
    - With research API enabled
    - With 3D-printed markers attached on the trocars of ECM or PSMs
    - Need an NDA in place with Intuitive Surgical Inc.
3. A reasonably powerful Linux PC
    - With binaries of the da Vinci Si research API
    - With DeckLink Duo (2) frame grabbing card, connected with the two channels of SDI outputs from da Vinci Si

- With NVIDIA graphics card that supports CUDA version higher than 10.1 and supports NVENC
- With custom-built FFmpeg libraries that support DeckLink input format, and nvidia library suites
- With cisst-saw (https://github.com/jhu-cisst/cisst-saw) installed, including sawIntuitiveDaVinci (https://github.com/jhu-saw/sawIntuitiveDaVinci) and sawSocketStreamer (https://github.com/jhu-saw/sawSocketStreamer)
- Suggested version: Ubuntu 18.04

4. A router
   - Configured to use 10.0.0.1 as host IP address, and make sure that 10.0.0.5 is not assigned by DHCP
   - Connected with da Vinci Si using Ethernet cable
   - Connected with the Linux PC using Ethernet cable
   - Wirelessly connected with HoloLens v1
5. A Windows 10 PC for development
   - With Unity 2018.3.XX
   - With Windows 10 SDK
   - With Visual Studio 2015 or higher, with Common Tools for Visual C++ feature installed
   - Paired with HoloLens v1 that enables deployment

2 Compile *ARssist*

The compilation procedure is listed as in the following subsections.

2.1 On Windows PC

1. **OPTIONAL**: Compile FFmpeg libraries for UWP
    - UWP is the OS that HoloLens runs on. In order to use ffmpeg functionalities, it has to be compatible with the OS. This compiles the ffmpeg development libraries for UWP platforms.
    - Follow the guide on https://trac.ffmpeg.org/wiki/CompilationGuide/WinRT (Compiling for Windows 10). Note that the FFmpeg configuration option `-disable-d3d11va` should be replaced with `-enable-d3d11va`, so that hardware acceleration using D3D11 is supported.
    - The results include the header files and libraries (e.g. `avformat.dll`) that are needed to compile `FFmpegUnityInterop.dll`
2. **OPTIONAL**: Compile `FFmpegUnityInterop.dll`
    - Clone the project at https://git.lcsr.jhu.edu/aramis/ffmpegunityinterop
    - Open the solution file with Visual Studio 2015 or higher
    - Make sure that the folder `FFmpegUnityInterop/ffmpeg/` includes the header files and libraries files of ffmpeg that are compatible with UWP. The Visual

Studio project is dependent on them. If the developer builds a customized version of ffmpeg libraries, please paste them in this folder.

- Build the solution, then `FFmpegUnityInterop.dll` should appear in the folder `Release/FFmpegUnityInterop/`.

3. Compile Unity application `ARssistSi`

   - Clone the project at https://git.lcsr.jhu.edu/aramis/hololensarssistsi
   - If you followed step 1 and 2, please copy the following library files to the `Assets/FFmpegUnityInterop/Plugins/WSA/x86/` path in the Unity project: `FFmpegUnityInterop.dll`, `avcodec-57.dll`, `avdevice-57.dll`, `avfilter-6.dll`, `avformat-57.dll`, `avutil-55.dll`, `swresample-2.dll` and `swscale-4.dll`.
   - Open the Unity project with Unity editor 2018.3.XX
   - Select Unity scene file `ARssistSi`, and build the Visual Studio solution file from it. Make sure that `Universal Windows Platform` is chosen as the target platform. The general settings for HoloLens project should be applied (see Build and deploy to device from Visual Studio).
   - Open the Visual Studio solution file, change the build target to `Release/x86`, then build and deploy to `Device`. HoloLens should be connected to the PC via USB. Wireless deployment is also OK.

2.2 On Linux PC

1. Compile FFmpeg libraries with DeckLink and nvenc support
    - Make sure CUDA is installed, version 10.1 or newer
    - Make sure DeckLink SDK is installed
    - Follow https://trac.ffmpeg.org/wiki/CompilationGuide/Ubuntu as a general guide for compilation of FFmpeg for Ubuntu.
    - Turn on the DeckLink support by adding `-enable-decklink` configuration option, and provide the path to the include directory of DeckLink BMD SDK. You can refer to gist: ffmpeg with decklink for more details.
    - Turn on the CUDA support, please follow https://devblogs.nvidia.com/nvidia-ffmpeg-transcoding-guide/.
    - Compile and install the FFmpeg libraries on the Linux PC.
    - If the official binaries support DeckLink and nvenc, you can also use the official binaries.
2. Compile `DeckLinkStream`
    - Clone the project at https://git.lcsr.jhu.edu/aramis/decklinkstream
    - Modify the IP address in `ffmpeg_decklink_stereo_streamer.cpp` as the HoloLens IP in the same subnet.
    - Create a `Build` folder and use CMake to compile the program.

- `ffmpeg_decklink_stereo_streamer` is the main executable for streaming the stereo endoscopic video to HoloLens.

3. Compile `sawIntuitiveDaVinci`

- This application streams the kinematics of the da Vinci Si robot to HoloLens.
- Follow https://github.com/jhu-cisst/cisst/wiki/Compiling-cisst-and-SAW-with-CMake to compile the `cisst` libraries and `cisst-saw` suites.
- With the latest version, `sawSocketStreamer` should be included.
- A few configuration files are needed to run `sawIntuitiveDaVinci` with streaming capability. They are listed in `share/socket-streamer` folder. Run `sawIntuitiveDaVinci` with the option to add a JSON manager, using `-m manager-socket-streamer-patient-cart.json`, which includes three `sawSocketStreamer` streams, corresponding to the kinematics of ECM, PSM1 and PSM2. The rate of streaming can be configured in the manager file. And IP addresses and port of each individual stream is configured in `streamerXXX.json`.

3 Run *ARssist*

Three applications are needed to run *ARssist*:

1. The HoloLens application `ARssistSi`

2. The Linux application `ffmpeg_decklink_stereo_streamer`
3. The Linux application `sawIntuitiveDaVinci`

The order to start each application does not matter for *ARssist*. It is important to make sure:

1. The Linux PC, da Vinci Si, HoloLens are under the same network.
2. The DeckLink Duo (2) are connected with the two 720P SDI channels of da Vinci Si. Use the BlackMagic SDK application to verify the connection is established.
3. IP address of `ffmpeg_decklink_stereo_streamer` is correctly set to the HoloLens address.
4. IP address of `sawIntuitiveDaVinci`, specified in the files `streamerXXX.json`, are correctly set to HoloLens address.
5. The fiducial markers are correctly affixed to the trocars.
6. It is suggested to run `sendKey.py` Python script on the Linux PC that sends commands to *ARssist*, more details will be listed in the subsection below.

When all applications are running, press the key 's' and then 'v' in the python script to start the video decoding and visualization on HoloLens.

3.1 Keyboard Commands

Run the Python 3 script `sendKey.py`, make sure the IP address is set to the HoloLens address, then Unity application `ARssistSi` can be controlled via keyboard

inputs. The keys and functions are:

1. 's': Start decoder
2. 't': Pause or resume fiducial tracking
3. 'd': Start or stop debugging mode
4. 'r': Switch to next visualization mode of endoscopic video, iterating between i) none, ii) heads-up display, iii) virtual monitor, and iv) frustum projection.
5. 'z': Switch visualization mode to none
6. 'x': Switch visualization mode to heads-up display
7. 'c': Switch visualization mode to virtual monitor
8. 'v': Switch visualization mode to frustum projection
9. 'u': In virtual monitor mode, move back the virtual monitor
10. 'i': In virtual monitor mode, move forward the virtual monitor
11. 'o': In virtual monitor mode, scale up the virtual monitor
12. 'p': In virtual monitor mode, scale down the virtual monitor
13. 'n': In heads-up display mode, move back the display
14. 'm': In heads-up display mode, move forward the display

4 Notices

Please check the repository wiki for errata and updates to these instructions. Consult me (lqian8@jhu.edu) when running into problems.

Appendix C

How to Compile *ARAMIS*

This appendix details the compilation procedure of *ARAMIS*, an AR application on Microsoft HoloLens v1 implemented with Unity. The methods and evaluations of *ARAMIS* are presented in Ch. 6. The developer is assumed to have good understanding of computer science, programming experience with Windows and Linux, and development experience with Unity and HoloLens. The following instructions are for using *ARAMIS* with the stereo endoscope of a da Vinci Si robot and rely on the research interface. *ARAMIS* could be used in other configurations, such as with dVRK or other stereo endoscope.

1 Requirements

First, the developer needs to access the repositories on the LCSR internal git server (https://git.lcsr.jhu.edu/aramis), including:

1. PointCloudInterop: https://git.lcsr.jhu.edu/aramis/pointcloudinterop
2. HoloLensARAMIS: https://git.lcsr.jhu.edu/aramis/hololensaramis
3. DeckLinkReconstruction: https://git.lcsr.jhu.edu/aramis/decklinkreconstruction

1.1 Requirements and Setup for Compilation

The developer needs the following hardware and software (the version information is the one I am using, other versions not guaranteed to work):

1. Microsoft HoloLens v1
    - With developing mode, not necessarily research mode
    - OS version: 10.0.10240.0 to 10.0.17763.0
2. da Vinci Si
    - With research API enabled
    - With 3D-printed markers attached on the endoscope (e.g., the trocar of the da Vinci ECM)
    - Need an NDA in place with Intuitive Surgical Inc.
3. A reasonably powerful Linux PC

- With binaries of the da Vinci Si research API
- With DeckLink Duo (2) frame grabbing card, connected with the two channels of SDI outputs from da Vinci Si
- With NVIDIA graphics card that supports CUDA version higher than 10.1
- With OpenCV installed with CUDA support
- With cisst-saw (https://github.com/jhu-cisst/cisst-saw) installed, including sawIntuitiveDaVinci (https://github.com/jhu-saw/sawIntuitiveDaVinci) and sawSocketStreamer (https://github.com/jhu-saw/sawSocketStreamer)
- Suggested version: Ubuntu 18.04

4. A router
   - Configured to use 10.0.0.1 as host IP address, and make sure that 10.0.0.5 is not assigned by DHCP
   - Connected with da Vinci Si using Ethernet cable
   - Connected with the Linux PC using Ethernet cable
   - Wirelessly connected with HoloLens v1
5. A Windows 10 PC for development
   - With Unity 2018.3.XX
   - With Windows 10 SDK
   - With Visual Studio 2015 or higher

- Paired with HoloLens v1 that enables deployment

2 Compile *ARAMIS*

The compilation procedure is listed as in the following subsections.

2.1 On Windows PC

1. **OPTIONAL**: Compile `PointCloudInterop.dll`
   - Clone the project at https://git.lcsr.jhu.edu/aramis/pointcloudinterop
   - Open the solution file with Visual Studio 2015 or higher
   - Build the project `PointCloudInterop`, then `PointCloudInterop.dll` should appear in the folder `Release/PointCloudInterop/`.
2. Compile Unity application `ARAMIS`
   - Clone the project at https://git.lcsr.jhu.edu/aramis/hololensaramis
   - If you followed step 1, please copy `PointCloudInterop.dll` to the folder `Assets/PointCloud/Plugins/WSA/x86/` in the Unity project.
   - Open the Unity project with Unity editor 2018.3.XX
   - Select Unity scene file `ARAMIS`, and build the Visual Studio solution file from it. Make sure that `Universal Windows Platform` is chosen as the

target platform. The general settings for the HoloLens project should be applied (see Build and deploy to device from Visual Studio).

- Open the Visual Studio solution file, change the build target to `Release/x86`, then build and deploy to `Device`. HoloLens should be connected to the PC via USB. Wireless deployment is also OK.

2.2 On Linux PC

1. Compile dependency `libSGM`
   - This library implements the core functionality of disparity calculation using CUDA
   - Make sure CUDA is installed, version 10.1 or newer. Make sure OpenCV is installed with CUDA support
   - Clone the project at https://github.com/fixstars/libSGM
   - Build and install on the Linux PC
2. Compile `DeckLinkReconstruction`
   - Clone the project at https://git.lcsr.jhu.edu/aramis/decklinkreconstruction
   - Modify the IP address in `main.cpp` as the HoloLens IP in the same subnet.
   - Create a `Build` folder and use CMake to compile the program. The program depends on `libSGM`, OpenCV, Boost and DeckLink SDK

- `DeckLinkReconstruction` is the main executable for streaming the point cloud to HoloLens.

3. Compile `sawIntuitiveDaVinci`

   - This application streams the kinematics of the da Vinci Si robot to HoloLens.
   - Follow https://github.com/jhu-cisst/cisst/wiki/Compiling-cisst-and-SAW-with-CMake to compile the `cisst` libraries and `cisst-saw` suites.
   - With the latest version, `sawSocketStreamer` should be included.
   - A few configuration files are needed to run `sawIntuitiveDaVinci` with streaming capability. They are listed in `share/socket-streamer` folder. Run `sawIntuitiveDaVinci` with the option to add a JSON manager, using `-m manager-socket-streamer-aramis.json`, which includes one `sawSocketStreamer` stream, corresponding to the kinematics of ECM. The rate of streaming can be configured in the manager file. IP addresses and port are configured in `streamerECM1.json`.

3 Run *ARAMIS*

Three applications are needed to run *ARAMIS*:

1. The HoloLens application `ARAMIS`
2. The Linux application `DeckLinkReconstruction`

3. The Linux application `sawIntuitiveDaVinci`

The order to start each application does not matter for *ARAMIS*. It is important to make sure:

1. The Linux PC, da Vinci Si, HoloLens are under the same network.
2. The DeckLink Duo (2) are connected with the two 720P SDI channels of da Vinci Si. Use the BlackMagic SDK application to verify the connection is established.
3. IP address of `DeckLinkReconstruction` is correctly set to HoloLens address.
4. IP address of `sawIntuitiveDaVinci`, specified in the file `streamerECM1.json` is correctly set to HoloLens address
5. The fiducial markers are correctly affixed to the trocars.
6. It is suggested to run the `sendKey.py` Python script on the Linux PC that sends command to *ARAMIS*, more details will be listed in the subsection below.

When all applications are running, the user should see the point cloud start playing registered with the endoscope.

3.1 Keyboard Commands

Run the Python 3 script `sendKey.py`, make sure the IP address is set to the HoloLens address, then Unity application ***`ARAMIS`*** can be controlled via keyboard inputs. The keys and functions are:

1. 't': Pause or resume fiducial tracking

2. 'd': Start or stop debugging mode
3. 'p': Toggle plane mode, where the point cloud depth channel is set to a fixed value, and the point cloud is rendered as a plane.

4 Notices

When the point cloud does not seem correct, or contains a lot of noise, you may need to recalibrate the stereo endoscope. Place a checkerboard beneath the endoscope with various poses, and capture the left and right channels from the endoscope. Use the Matlab stereo camera calibration app to calibrate them, which will generate a `config.yaml` file. You may manually convert it to OpenCV standards. Examples are provided in `DeckLinkReconstruction/config` folder. Another way to quickly verify whether a re-calibration is needed is to change the focus of the endoscope using the buttons on the endoscope and see if the point cloud quality is best at a certain focal distance.

Please check the repository wiki for errata and updates to these instructions. Consult me (lqian8@jhu.edu) when running into problems.

Appendix D

Tips for Writing OST-HMD Programs using Unity

Throughout my Ph.D studies, I have developed many applications on OST-HMDs, including HoloLens, ODG glasses, and Moverio glasses. I would like to share a few tips about how to write efficient AR applications using Unity.

Most importantly, use Unity as a top-level framework to manage different tasks and resources. Unity is easy to learn and use. It wraps up low-level graphics, communication, system APIs and makes the tasks for developers easier. However, it is still a rendering engine after all, and it has its own "clock", the display. For an AR/VR platform, it means the Unity game logic functions, e.g. `Update()`, run at $60\,Hz$ or $75\,Hz$ depending on the hardware. Although there are many workarounds to enable functions to run on separate "clock", it still feels hacky and the code gets hard to man-

age. My suggestion is to use more native programming, and exchange data with the main function via CPU or via GPU. For HoloLens, I implement point cloud receiving (in *ARAMIS*), h264 stream decoding (in *ARssist*), fiducial tracking (in HoloLensARToolKit) in C++. The core functionalities are wrapped up in dynamic link libraries (DLL), to be called by the Unity application. The API needs to be defined explicitly for both the native library and the Unity C# script. In this way, Unity scripts will not be overwhelmed by the length of code of core functions. The core functions are encapsulated well, and the resource management is done separately. Also, when it comes to complex math computation, use native libraries! You can find all kinds of math libraries in C++.

The data exchange between the Unity project and native libraries is critical to the performance of the application. The Unity project uses managed memory, meaning that unused memory will be automatically Garbage Collected by the system. However, most native libraries are implemented using unmanaged memory. Therefore, the data need to be copied (Marshaling) between the managed-unmanaged border, each side keeping its own version of the same data. It is important to limit the number and size of memory copies. The above works for CPU data, and in some other cases, data exchange is on the GPU. It gets trickier when both the native library and Unity are handling the same GPU memory. Unity does provide an interface to allow native code to execute on the rendering loop, with more details at https://docs.unity3d.com/Manual/NativePluginInterface.html. The native library

can register functions on the Unity execution loop, and implement the GPU memory manipulation in those functions. In this way, the GPU resource will not be accessed at the same time by two parties.

Thirdly, it is recommended to create threads and run computationally expensive functions in other threads, e.g. fiducial tracking. The purpose is to not slow down the main rendering loop of Unity. Most of the expensive functions can run at a different rate compared to the visualization. For example, the tracking of objects can come up with a result at 10 frames per second, while Unity visualizes the virtual object registered with the tracked real object at $60\,Hz$. It is a very different user experience if the visualization loop is bounded by the tracking result, so that both of them run at $10\,Hz$. The user will immediately dislike the application. Again, when the native functions are not running at the same rate as the Unity main application, the data exchange needs to be carefully managed. Think of it as exchanging data from two different processes. The data dependency needs to be loose.

Lastly, it is also important for developers to make sure that users actually received the desired experience. For new users, it means we need to help them wear the headset correctly, without putting too much weight on the nose. Otherwise, the users will hate the weight of the headset, and cannot positively perceive what you want to demo to them. It takes at most 2 minutes to make sure they wear the headset correctly, but it makes huge difference.

May your AR application run at $60\,Hz$ and be WOW-ed by others!!!

Bibliography

[1] D. F. Abawi, J. Bienwald, and R. Dorner. Accuracy in optical tracking with fiducial markers: an accuracy function for ARToolKit. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 260–261. IEEE Computer Society, 2004.

[2] R. W. Adams. *Peripheral vision and visual attention.* PhD thesis, Iowa State University, 1971.

[3] T. K. Adebar, M. C. Yip, S. E. Salcudean, R. N. Rohling, C. Y. Nguan, and S. L. Goldenberg. Registration of 3D ultrasound through an air–tissue boundary. *TMI*, 31(11):2133–2142, 2012.

[4] J. Afthinos, M. Latif, F. Bhora, C. Connery, J. McGinty, A. Burra, M. Attiyeh, G. Todd, and S. Belsley. What technical barriers exist for real-time fluoroscopic and video image overlay in robotic surgery? *IJMRCAS*, 4(4):368–372, 2008.

[5] T. Akinbiyi, C. E. Reiley, S. Saha, D. Burschka, C. J. Hasser, D. D. Yuh,

and A. M. Okamura. Dynamic augmented reality for sensory substitution in robot-assisted surgical systems. In *EMBS*, pages 567–570. IEEE, 2006.

[6] S. Alletto, G. Serra, and R. Cucchiara. Egocentric Object Tracking: An Odometry-Based Solution. In *Proceedings of the International Conference on Image Analysis and Processing*, pages 687–696, 2015.

[7] A. Amir-Khalili, M. S. Nosrati, J.-M. Peyrat, G. Hamarneh, and R. Abugharbieh. Uncertainty-encoded augmented reality for robot-assisted partial nephrectomy: A phantom study. In *Augmented Reality Environments for Medical Imaging and Computer-Assisted Interventions*, pages 182–191. Springer, 2013.

[8] B. J. Ardila, M. A. Orvieto, and V. R. Patel. Role of the robotic surgical assistant. In *Robotic Urologic Surgery*, pages 495–505. Springer, 2011.

[9] J. Ardouin, A. Lécuyer, M. Marchal, C. Riant, and E. Marchand. FlyVIZ: A Novel Display Device to Provide Humans with 360 Vision by Coupling Catadioptric Camera with HMD. In *Proceedings of the ACM symposium on Virtual Reality Software and Technology*, pages 41–44, 2012.

[10] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-D point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.

[11] M. Axholt, S. Peterson, and S. R. Ellis. User boresight calibration precision for

large-format head-up displays. In *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*, pages 141–148. ACM, 2008.

[12] M. Axholt, S. D. Peterson, and S. R. Ellis. Visual alignment precision in optical see-through AR displays: Implications for potential accuracy. In *Proceedings of the ACM/IEEE Virtual Reality International Conference*, 2009.

[13] M. Axholt, M. Skoglund, S. D. Peterson, M. D. Cooper, T. B. Schön, F. Gustafsson, A. Ynnerman, and S. R. Ellis. Optical see-through head mounted display direct linear transformation calibration robustness in the presence of user alignment noise. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 54, pages 2427–2431. SAGE Publications Sage CA: Los Angeles, CA, 2010.

[14] M. Axholt, M. A. Skoglund, S. D. O'Connell, M. D. Cooper, S. R. Ellis, and A. Ynnerman. Parameter estimation variance of the single point active alignment method in optical see-through head mounted display calibration. In *Virtual Reality Conference (VR), 2011 IEEE*, pages 27–34. IEEE, 2011.

[15] M. Azizian and R. Patel. Intraoperative 3D stereo visualization for image-guided cardiac ablation. In *Medical Imaging 2011: Visualization, Image-Guided Procedures, and Modeling*, volume 7964, page 79640F. International Society for Optics and Photonics, 2011.

BIBLIOGRAPHY

[16] R. T. Azuma. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, 6(4):355–385, 1997.

[17] M. Bajura, H. Fuchs, and R. Ohbuchi. Merging virtual objects with the real world: Seeing ultrasound imagery within the patient. *ACM SIGGRAPH Computer Graphics*, 26(2):203–210, 1992.

[18] M. A. Balicki. *Augmentation of human skill in microsurgery.* PhD thesis, Johns Hopkins University, 2014.

[19] A. Bangor, P. Kortum, and J. Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[20] J. M. Baste, V. Soldea, S. Lachkar, P. Rinieri, M. Sarsam, B. Bottet, and C. Peillon. Development of a precision multimodal surgical navigation system for lung robotic segmentectomy. *Journal of Thoracic Disease*, 10(Suppl 10):S1195, 2018.

[21] P. Baudisch and R. Rosenholtz. Halo: A Technique for Visualizing Off-Screen Objects. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 481–488, 2003.

[22] S. Bernhardt, J. Abi-Nahed, and R. Abugharbieh. Robust dense endoscopic stereo reconstruction for minimally invasive surgery. In *MICCAI Workshop on Medical Computer Vision*, pages 254–262. Springer, 2012.

[23] C. Bichlmeier, F. Wimmer, S. M. Heining, and N. Navab. Contextual anatomic mimesis hybrid in-situ visualization method for improving multi-sensory depth perception in medical augmented reality. In *ISMAR*, pages 129–138. IEEE, 2007.

[24] A. Bihlmaier, T. Beyl, P. Nicolai, M. Kunze, J. Mintenbeck, L. Schreiter, T. Brennecke, J. Hutzl, J. Raczkowsky, and H. Wörn. ROS-based cognitive surgical robotics. In *Robot Operating System (ROS)*, pages 317–342. Springer, 2016.

[25] F. Biocca, A. Tang, C. Owen, and F. Xiao. Attention Funnel: Omnidirectional 3D Cursor for Mobile Augmented Reality Platforms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1115–1122, 2006.

[26] F. A. Biocca and J. P. Rolland. Virtual Eyes Can Rearrange Your Body: Adaptation to Visual Displacement in See-Through, Head-Mounted Displays. *Presence*, 7(3):262–277, 1998.

[27] W. Birkfellner, M. Figl, K. Huber, F. Watzinger, F. Wanschitz, R. Hanel, A. Wagner, D. Rafolt, R. Ewers, and H. Bergmann. The varioscope ar–a head-mounted operating microscope for augmented reality. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 869–877. Springer, 2000.

[28] W. Birkfellner, K. Huber, F. Watzinger, M. Figl, F. Wanschitz, R. Hanel, D. Rafolt, R. Ewers, and H. Bergmann. Development of the varioscope ar. a see-through hmd for computer-aided surgery. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*, pages 54–59. IEEE, 2000.

[29] R. Bogdanova, P. Boulanger, and B. Zheng. Depth perception of surgeons in minimally invasive surgery. *Surgical innovation*, 23(5):515–524, 2016.

[30] S. Burigat and L. Chittaro. Navigation in 3D Virtual Environments: Effects of User Experience and Location-Pointing Navigation Aids. *International Journal of Human-Computer Studies*, 65(11):945–958, 2007.

[31] S. Burigat, L. Chittaro, and S. Gabrielli. Visualizing Locations of Off-Screen Objects on Mobile Devices: A Comparative Evaluation of Three Approaches. In *Proceedings of the Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 239–246, 2006.

[32] D. B. Camarillo, T. M. Krummel, and J. K. Salisbury Jr. Robotic technology in surgery: past, present, and future. *The American Journal of Surgery*, 188(4):2–15, 2004.

[33] M.-A. Cardin, J.-X. Wang, and D. B. Plewes. A method to evaluate human spatial coordination interfaces for computer-assisted surgery. In *International*

Conference on Medical Image Computing and Computer-Assisted Intervention, pages 9–16. Springer, 2005.

[34] P.-L. Chang, D. Stoyanov, A. J. Davison, et al. Real-time dense stereo reconstruction using convex optimisation with a cost-volume for image-guided robotic surgery. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 42–49. Springer, 2013.

[35] C. B. Chen. Wide Field of View, Wide Spectral Band Off-Axis Helmet-Mounted Display Optical Design. In *Proceedings of the International Optical Design Conference*, volume 4832, pages 61–67, 2002.

[36] E. C. Chen, K. Sarkar, J. S. Baxter, J. Moore, C. Wedlake, and T. M. Peters. An augmented reality platform for planning of minimally invasive cardiac surgeries. In *Medical Imaging 2012: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 8316, page 831617. International Society for Optics and Photonics, 2012.

[37] X. Chen, L. Xu, Y. Wang, H. Wang, F. Wang, X. Zeng, Q. Wang, and J. Egger. Development of a surgical navigation system based on augmented reality using an optical see-through head-mounted display. *Journal of biomedical informatics*, 55:124–131, 2015.

[38] C. P. Childers and M. Maggard-Gibbons. Understanding costs of care in the operating room. *JAMA Surgery*, 153(4):e176233–e176233, 2018.

[39] P. C. Chimenti and D. J. Mitten. Google glass as an alternative to standard fluoroscopic visualization for percutaneous fixation of hand fractures: a pilot study. *Plastic and reconstructive surgery*, 136(2):328–330, 2015.

[40] K. Chintamani, A. Cao, R. D. Ellis, C.-A. Tan, and A. K. Pandya. An analysis of teleoperator performance in conditions of display-control misalignments with and without movement cues. *Journal of Cognitive Engineering and Decision Making*, 5(2):139–155, 2011.

[41] W. Chou, T. Wang, and Y. Zhang. Augmented reality based preoperative planning for robot assisted tele-neurosurgery. In *International Conference on Systems, Man and Cybernetics*, volume 3, pages 2901–2906. IEEE, 2004.

[42] N. H. Christensen, O. G. Hjermitslev, F. Falk, M. B. Madsen, F. H. Østergaard, M. Kibsgaard, M. Kraus, J. Poulsen, and J. Petersson. Depth cues in augmented reality for training of robot-assisted minimally invasive surgery. In *International Academic Mindtrek Conference*, pages 120–126. ACM, 2017.

[43] W. Clark, P. Bird, P. Gonski, T. H. Diamond, P. Smerdely, H. P. McNeil, G. Schlaphoff, C. Bryant, E. Barnes, and V. Gebski. Safety and efficacy of vertebroplasty for acute painful osteoporotic fractures (vapour): a multicentre, randomised, double-blind, placebo-controlled trial. *The Lancet*, 388(10052):1408–1416, 2016.

[44] Colon Cancer Laparoscopic or Open Resection Study Group. Laparoscopic

surgery versus open surgery for colon cancer: short-term outcomes of a randomised trial. *The lancet oncology*, 6(7):477–484, 2005.

[45] E. Costanza, S. A. Inverso, E. Pavlov, R. Allen, and P. Maes. Eye-q: Eyeglass peripheral display for subtle intimate notifications. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 211–218. ACM, 2006.

[46] È. Coste-Manière, L. Adhami, F. Mourgues, and O. Bantiche. Optimal planning of robotically assisted heart surgery: first results on the transfer precision in the operating room. *IJRR*, 23(4-5):539–548, 2004.

[47] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: the design and implementation of the cave. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 135–142, 1993.

[48] C. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson. Human Photoreceptor Topography. *Journal of Comparative Neurology*, 292(4):497–523, 1990.

[49] F. Danieau, A. Guillo, and R. Doré. Attention guidance for immersive video content in head-mounted displays. In *Virtual Reality (VR), 2017 IEEE*, pages 205–206. IEEE, 2017.

[50] B. Davies, F. Rodriguez y Baena, A. Barrett, M. Gomes, S. Harris, M. Jakopec, and J. Cobb. Robotic control in knee joint replacement surgery. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 221(1):71–80, 2007.

[51] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1052–1067, 2007.

[52] E. S. De Guzman, M. Yau, A. Gagliano, A. Park, and A. K. Dey. Exploring the design and use of peripheral displays of awareness information. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 1247–1250. ACM, 2004.

[53] T. De Silva, J. Punnoose, A. Uneri, J. Goerres, M. Jacobson, M. D. Ketcha, A. Manbachi, S. Vogt, G. Kleinszig, A. J. Khanna, et al. C-arm positioning using virtual fluoroscopy for image-guided surgery. In *Medical Imaging 2017: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 10135, page 101352K. International Society for Optics and Photonics, 2017.

[54] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. IEEE, 2009.

[55] A. M. Derossis, G. M. Fried, H. H. Sigman, J. S. Barkun, and J. L. Meakins.

Development of a model for training and evaluation of laparoscopic skills 1. *The American Journal of Surgery*, 175(6):482–487, 1998.

[56] F. Devernay, F. Mourgues, and È. Coste-Manière. Towards endoscopic augmented reality for robotically assisted minimally invasive cardiac surgery. In *MIAR*, pages 16–20. IEEE, 2001.

[57] P. Edgcumbe, P. Pratt, G.-Z. Yang, C. Nguan, and R. Rohling. Pico lantern: surface reconstruction and augmented reality in laparoscopic surgery using a pick-up laser projector. *Medical Image Analysis*, 25(1):95–102, 2015.

[58] P. Edgcumbe, R. Singla, P. Pratt, C. Schneider, C. Nguan, and R. Rohling. Augmented reality imaging for robot-assisted partial nephrectomy surgery. In *International Conference on Medical Imaging and Virtual Reality*, pages 139–150. Springer, 2016.

[59] P. Edgcumbe, R. Singla, P. Pratt, C. Schneider, C. Nguan, and R. Rohling. Follow the light: projector-based augmented reality intracorporeal system for laparoscopic surgery. *JMI*, 5(2):021216, 2018.

[60] K. Erfanian, F. I. Luks, A. G. Kurkchubasche, C. W. Wesselhoeft Jr, and T. F. Tracy Jr. In-line image projection accelerates task performance in laparoscopic appendectomy. *Journal of pediatric surgery*, 38(7):1059–1062, 2003.

[61] O. Fachklinik, S. Schwarzach, C. G. Beaulieu, H. Stoevelaar, and L. Belgium.

Treatment of osteoporotic vertebral compression fractures: applicability of appropriateness criteria in clinical practice. *Pain Phys*, 19:E113–20, 2016.

[62] V. Falk, F. Mourgues, L. Adhami, S. Jacobs, H. Thiele, S. Nitzsche, F. W. Mohr, and È. Coste-Manière. Cardio navigation: planning, simulation, and augmented reality in robotic assisted endoscopic bypass grafting. *The Annals of Thoracic Surgery*, 79(6):2040–2047, 2005.

[63] K. Fan, J. Huber, S. Nanayakkara, and M. Inami. SpiderVision: Extending the Human Field of View for Augmented Awareness. In *Proceedings of the Augmented Human International Conference*, pages 49:1–49:8, 2014.

[64] G. Farnebäck. Two-Frame Motion Estimation Based on Polynomial Expansion. In *Proceedings of the Scandinavian Conference on Image Analysis*, pages 363–370, 2003.

[65] S. Feiner, B. Macintyre, and D. Seligmann. Knowledge-based augmented reality. *Communications of the ACM*, 36(7):53–62, 1993.

[66] M. Feuerstein, T. Mussack, S. M. Heining, and N. Navab. Intraoperative laparoscope augmentation for port placement and resection planning in minimally invasive liver resection. *T-MI*, 27(3):355–369, 2008.

[67] M. Figl, C. Ede, J. Hummel, F. Wanschitz, R. Ewers, H. Bergmann, and W. Birkfellner. A fully automated calibration method for an optical see-through

head-mounted operating microscope with variable zoom and focus. *IEEE transactions on medical imaging*, 24(11):1492–1499, 2005.

[68] M. Figl, D. Rueckert, D. Hawkes, R. Casula, M. Hu, O. Pedro, D. P. Zhang, G. Penney, F. Bello, and P. Edwards. Augmented reality image guidance for minimally invasive coronary artery bypass. In *Medical Imaging 2008: Visualization, Image-Guided Procedures, and Modeling*, volume 6918, page 69180P. International Society for Optics and Photonics, 2008.

[69] G. Fontanelli, F. Ficuciello, L. Villani, and B. Siciliano. Da vinci research kit: Psm and mtm dynamic modelling. In *IROS Workshop on Shared Platforms for Medical Robotics Research, Vancouver, Canada*, 2017.

[70] H. Fuchs, M. A. Livingston, R. Raskar, K. Keller, J. R. Crawford, P. Rademacher, S. H. Drake, A. A. Meyer, et al. Augmented reality visualization for laparoscopic surgery. In *MICCAI*, pages 934–943. Springer, 1998.

[71] B. Fuerst, J. Sprung, F. Pinto, B. Frisch, T. Wendler, H. Simon, L. Mengus, N. S. van den Berg, H. G. van der Poel, F. W. van Leeuwen, et al. First robotic SPECT for minimally invasive sentinel lymph node mapping. *TMI*, 35(3):830–838, 2016.

[72] A. Fuhrmann, D. Schmalstieg, and W. Purgathofer. Fast calibration for augmented reality. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 166–167. ACM, 1999.

[73] A. Fuhrmann, D. Schmalstieg, and W. Purgathofer. Practical calibration procedures for augmented reality. In *Virtual Environments 2000*, pages 3–12. Springer, 2000.

[74] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

[75] Y. Genc, F. Sauer, F. Wenzel, M. Tuceryan, and N. Navab. Optical see-through hmd calibration: A stereo method validated with a video see-through system. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 165–174. IEEE, 2000.

[76] Y. Genc, M. Tuceryan, A. Khamene, and N. Navab. Optical see-through calibration with vision-based trackers: Propagation of projection matrices. In *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*, pages 147–156. IEEE, 2001.

[77] Y. Genc, M. Tuceryan, and N. Navab. Practical solutions for calibration of optical see-through devices. In *Proceedings of the 1st International Symposium on Mixed and Augmented Reality*, page 169. IEEE Computer Society, 2002.

[78] D. B. George and L. R. Morris. A computer-driven astronomical telescope guidance and control system with superimposed star field and celestial coordi-

nate graphics display. *Journal of the Royal Astronomical Society of Canada*, 83:32–41, 1989.

[79] F. Gîrbacia, R. Boboc, B. Gherman, T. Gîrbacia, and D. Pîsla. Planning of needle insertion for robotic-assisted prostate biopsy in augmented reality using RGB-D camera. In *International Conference on Robotics in Alpe-Adria Danube Region*, pages 515–522. Springer, 2016.

[80] D. Gorpas, J. Phipps, J. Bec, D. Ma, S. Dochow, D. Yankelevich, J. Sorger, J. Popp, A. Bewley, R. Gandour-Edwards, L. Marku, and D. Farwell. Autofluorescence lifetime augmented reality as a means for real-time robotic surgery guidance in human patients. *Scientific Reports*, 9(1):1187, 2019.

[81] O. G. Grasa, J. Civera, and J. Montiel. EKF monocular SLAM with relocalization for laparoscopic sequences. In *ICRA*, pages 4816–4821. IEEE, 2011.

[82] R. Gregory and P. Cavanagh. The Blind Spot. *Scholarpedia*, 6(10):9618, 2011.

[83] J. Grubert, Y. Itoh, K. Moser, and J. E. Swan. A survey of calibration methods for optical see-through head-mounted displays. *IEEE transactions on visualization and computer graphics*, 24(9):2649–2662, 2017.

[84] J. Grubert, J. Tuemle, R. Mecke, and M. Schenk. Comparative user study of two see-through calibration methods. *VR*, 10:269–270, 2010.

[85] U. Gruenefeld, A. E. Ali, W. Heuten, and S. Boll. Visualizing Out-of-View Objects in Head-Mounted Augmented Reality. In *Proceedings of the International Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 81:1–81:7, 2017.

[86] U. Gruenefeld, D. Ennenga, A. E. Ali, W. Heuten, and S. Boll. EyeSee360: Designing a Visualization Technique for Out-of-View Objects in Head-Mounted Augmented Reality. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pages 109–118, 2017.

[87] U. Gruenefeld, D. Hsiao, W. Heuten, and S. Boll. EyeSee: Beyond Reality with Microsoft HoloLens. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pages 148–148, 2017.

[88] U. Gruenefeld, T. C. Stratmann, W. Heuten, and S. Boll. PeriMR: A Prototyping Tool for Head-Mounted Peripheral Light Displays in Mixed Reality. In *Proceedings of the International Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 51:1–51:6, 2017.

[89] S. Gustafson, P. Baudisch, C. Gutwin, and P. Irani. Wedge: Clutter-Free Visualization of Off-screen Locations. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 787–796, 2008.

[90] G. Guthart and J. Salisbury. The Intuitive^{TM} telesurgery system: Overview and application. In *ICRA*, pages 618–621, May 2000.

[91] R. J. Hallifax, J. P. Corcoran, A. Ahmed, M. Nagendran, H. Rostom, N. Hassan, M. Maruthappu, I. Psallidas, A. Manuel, F. V. Gleeson, et al. Physician-based ultrasound-guided biopsy for diagnosing pleural disease. *Chest*, 146(4):1001–1006, 2014.

[92] G. Hamarneh, A. Amir-Khalili, M. S. Nosrati, I. Figueroa, J. Kawahara, O. Al-Alao, J.-M. Peyrat, J. Abi-Nahed, A. Al-Ansari, and R. Abugharbieh. Towards multi-modal image-guided tumour identification in robot-assisted partial nephrectomy. In *MECBME*, pages 159–162. IEEE, 2014.

[93] G. B. Hanna, S. M. Shimi, and A. Cuschieri. Task performance in endoscopic surgery is influenced by location of the image display. *Annals of surgery*, 227(4):481, 1998.

[94] B. Hannaford, J. K. Barral, E. Rephaeli, C. D. Ching, and V. S. Bajaj. Heads-up displays for augmented reality network in a medical environment, May 9 2017. US Patent 9,645,785.

[95] B. Hannaford, J. Rosen, D. W. Friedman, H. King, P. Roan, L. Cheng, D. Glozman, J. Ma, S. N. Kosari, and L. White. Raven-II: an open platform for surgical robotics research. *T-BME*, 60(4):954–959, 2012.

[96] N. Haouchine, J. Dequidt, I. Peterlik, E. Kerrien, M.-O. Berger, and S. Cotin. Towards an accurate tracking of liver tumors for augmented reality in robotic assisted surgery. In *ICRA*, 2014.

[97] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 50, pages 904–908. Sage Publications Sage CA: Los Angeles, CA, 2006.

[98] S. G. Hart and L. E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology*, volume 52, pages 139–183. Elsevier, 1988.

[99] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[100] A. Hattori, N. Suzuki, M. Hashizume, T. Akahoshi, K. Konishi, S. Yamaguchi, M. Shimada, and M. Hayashibe. A robotic surgery system (da Vinci) with image guided function–system architecture and cholecystectomy application. *Studies in Health Technology and Informatics*, 94:110–116, 2003.

[101] E. Hecht. *Optics*. Pearson, 2015.

[102] M. L. Heilig. Sensorama simulator. *US Patemt (3,050,870), Patented August*, 28, 1962.

[103] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE TPAMI*, 30(2):328–341, 2008.

[104] D. M. Hoffman, A. R. Girshick, K. Akeley, and M. S. Banks. Vergence–

accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of vision*, 8(3):33–33, 2008.

[105] B. K. Horn. Closed-form solution of absolute orientation using unit quaternions. *JOSA A*, 4(4):629–642, 1987.

[106] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[107] Y.-H. Huang, T.-C. Yu, P.-H. Tsai, Y.-X. Wang, W.-L. Yang, and M. Ouhyoung. Scope+: a stereoscopic video see-through augmented reality microscope. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, pages 33–34. ACM, 2015.

[108] T. Huber, E. Hadzijusufovic, C. Hansen, M. Paschold, H. Lang, and W. Kneist. Head-mounted mixed-reality technology during robotic-assisted transanal total mesorectal excision. *Diseases of the Colon & Rectum*, 62(2):258–261, 2019.

[109] R. Hussain, A. Lalande, R. Marroquin, K. B. Girum, C. Guigou, and A. B. Grayeli. Real-time augmented reality for ear surgery. In *MICCAI*, pages 324–331. Springer, 2018.

[110] M. H. Iqbal, A. Aydin, O. Brunckhorst, P. Dasgupta, and K. Ahmed. A review of wearable technology in medicine. *Journal of the Royal Society of Medicine*, 109(10):372–380, 2016.

[111] Y. Itoh and G. Klinker. Interaction-free calibration for optical see-through head-mounted displays based on 3d eye localization. In *3D User Interfaces (3DUI), 2014 IEEE Symposium on*, pages 75–82. IEEE, 2014.

[112] Y. Itoh and G. Klinker. Performance and sensitivity analysis of indica: Interaction-free display calibration for optical see-through head-mounted displays. In *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*, pages 171–176. IEEE, 2014.

[113] Y. Itoh and G. Klinker. Light-Field Correction for spatial Calibration of Optical See-Through Head-Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics*, 21(4):471–480, 2015.

[114] T. James and A. S. Gilmour. Magnifying loupes in modern dental practice: an update. *Dental update*, 37(9):633–636, 2010.

[115] A. L. Janin, D. W. Mizell, and T. P. Caudell. Calibration of head-mounted displays for augmented reality applications. In *Virtual Reality Annual International Symposium*, pages 246–255. IEEE, 1993.

[116] A. M. Jarc, A. A. Stanley, T. Clifford, I. S. Gill, and A. J. Hung. Proctors exploit three-dimensional ghost tools during clinical-like training scenarios: a preliminary study. *World Journal of Urology*, 35(6):957–965, 2017.

[117] C. A. Johnson and J. L. Keltner. Incidence of Visual Field Loss in 20,000

Eyes and its Relationship to Driving Performance. *Archives of Ophthalmology*, 101(3):371–375, 1983.

[118] B. R. Jones, H. Benko, E. Ofek, and A. D. Wilson. Illumiroom: peripheral projected illusions for interactive experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 869–878. ACM, 2013.

[119] D. Joseph Tan, F. Tombari, S. Ilic, and N. Navab. A versatile learning-based 3d temporal tracker: Scalable, robust, online. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 693–701, 2015.

[120] R. S. Kalawsky and R. S. Kalawsky. *The science of virtual reality and virtual environments: a technical, scientific and engineering reference on virtual environments*. Addison-Wesley Workingham, 1993.

[121] H. Kato and M. Billinghurst. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *IWAR*, pages 85–94. IEEE, 1999.

[122] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio. An open-source research kit for the da Vinci® surgical system. In *ICRA*, pages 6434–6439. IEEE, 2014.

[123] P. Kazanzides, B. D. Mittelstadt, B. L. Musits, W. L. Bargar, J. F. Zuhars,

B. Williamson, P. W. Cain, and E. J. Carbone. An integrated system for cementless hip replacement. *Engineering in Medicine and Biology Magazine*, 14(3):307–313, 1995.

[124] K. Keller, A. State, and H. Fuchs. Head mounted displays for medical use. *Journal of display technology*, 4(4):468–472, 2008.

[125] S. Khamis, S. Fanello, C. Rhemann, A. Kowdle, J. Valentin, and S. Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *European Conference on Computer Vision (ECCV)*, pages 573–590, 2018.

[126] M. Kibsgaard and M. Kraus. Measuring the latency of an augmented reality system for robot-assisted minimally invasive surgery. In *GRAPP*, volume 2, pages 321–326. SCITEPRESS, 2017.

[127] M. Klemm, H. Hoppe, and F. Seebacher. Non-parametric camera-based calibration of optical see-through glasses for augmented reality applications. In *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 273–274. IEEE, 2014.

[128] M. Klemm, F. Seebacher, and H. Hoppe. Non-parametric Camera-Based Calibration of Optical See-Through Glasses for AR Applications. In *Proceedings of the International Conference on Cyberworlds*, pages 33–40, 2016.

[129] A. Kolagunda, S. Sorensen, S. Mehralivand, P. Saponaro, W. Treible, B. Turk-

bey, P. Pinto, P. Choyke, and C. Kambhamettu. A mixed reality guidance system for robot assisted laparoscopic radical prostatectomy. In *MICCAI OR 2.0 CARE Workshop*, pages 164–174. Springer, 2018.

[130] F. L. Kooi and M. Mosch. Peripheral motion displays: tapping the potential of the visual periphery. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 50, pages 1604–1608. SAGE Publications Sage CA: Los Angeles, CA, 2006.

[131] J. Kowalczuk, E. Psota, and L. C. Pérez. Stereoscopic vision-based robotic manipulator extraction method for enhanced soft tissue reconstruction. *Studies in Health Technology and Informatics*, 184:235–241, 2013.

[132] G. Kramida. Resolving the vergence-accommodation conflict in head-mounted displays. *IEEE T Vis Comput Grap*, 22(7):1912–1931, 2016.

[133] B. Kress and T. Starner. A review of head-mounted displays (HMD) technologies and applications for consumer electronics. In *SPIE*, volume 8720, page 87200A, 2013.

[134] R. Kumar and A. K. Hemal. The "scrubbed surgeon" in robotic surgery. *World Journal of Urology*, 24(2):144–147, 2006.

[135] D. M. Kwartowitz, S. D. Herrell, and R. L. Galloway. Toward image-guided robotic surgery: determining intrinsic accuracy of the da vinci robot. *Inter-*

national Journal of Computer Assisted Radiology and Surgery, 1(3):157–165, 2006.

[136] A. R. Lanfranco, A. E. Castellanos, J. P. Desai, and W. C. Meyers. Robotic surgery: a current perspective. *Annals of Surgery*, 239(1):14, 2004.

[137] A. M. Larson and L. C. Loschky. The Contributions of Central Versus Peripheral Vision to Scene Gist Recognition. *Journal of Vision*, 9(10):6:1–6:16, 2009.

[138] D. Lee, H.-J. Kong, D. Kim, J. W. Yi, Y. J. Chai, K. E. Lee, and H. C. Kim. Preliminary study on application of augmented reality visualization in robotic thyroid surgery. *Annals of Surgical Treatment and Research*, 95(6):297–302, 2018.

[139] J. Lemcke, F. Al-Zain, S. Mutze, and U. Meier. Minimally invasive spinal surgery using nucleoplasty and the dekompressor tool: a comparison of two methods in a one year follow-up. *min-Minimally Invasive Neurosurgery*, 53(05/06):236–242, 2010.

[140] M. Lerotic, A. J. Chung, G. Mylonas, and G.-Z. Yang. Pq-space based non-photorealistic rendering for augmented reality. In *MICCAI*, pages 102–109. Springer, 2007.

[141] J. Leven, D. Burschka, R. Kumar, G. Zhang, S. Blumenkranz, X. D. Dai, M. Awad, G. D. Hager, M. Marohn, M. Choti, C. Hasser, and R. H. Taylor.

DaVinci canvas: a telerobotic surgical system with integrated, robot-assisted, laparoscopic ultrasound capability. In *MICCAI*, pages 811–818. Springer, 2005.

[142] H. Liao, N. Hata, S. Nakajima, M. Iwahara, I. Sakuma, and T. Dohi. Surgical navigation by autostereoscopic image overlay of integral videography. *IEEE T Inf Technol B*, 8(2):114–121, 2004.

[143] J. Lin, N. T. Clancy, and D. S. Elson. An endoscopic structured light system using multispectral detection. *IJCARS*, 10(12):1941–1950, 2015.

[144] L. Lin, Y. Shi, A. Tan, M. Bogari, M. Zhu, Y. Xin, H. Xu, Y. Zhang, L. Xie, and G. Chai. Mandibular angle split osteotomy based on a novel augmented reality navigation using specialized robot-assisted arms–a feasibility study. *Journal of Cranio-Maxillofacial Surgery*, 44(2):215–223, 2016.

[145] G. Lintern. Transfer of landing skill after training with supplementary visual cues. *Human Factors*, 22(1):81–88, 1980.

[146] W. P. Liu, M. Azizian, J. Sorger, R. H. Taylor, B. K. Reilly, K. Cleary, and D. Preciado. Cadaveric feasibility study of da Vinci Si–assisted cochlear implant with augmented visual navigation for otologic surgery. *JAMA Otolaryngology–Head & Neck Surgery*, 140(3):208–214, 2014.

[147] W. P. Liu, J. D. Richmon, J. M. Sorger, M. Azizian, and R. H. Taylor. Aug-

mented reality and cone beam CT guidance for transoral robotic surgery. *Journal of Robotic Surgery*, 9(3):223–233, 2015.

[148] X. Liu, A. Sinha, M. Ishii, G. D. Hager, A. Reiter, R. H. Taylor, and M. Unberath. Self-supervised learning for dense depth estimation in monocular endoscopy. *ArXiv*, 2019.

[149] X. Liu, A. Sinha, M. Unberath, M. Ishii, G. D. Hager, R. H. Taylor, and A. Reiter. Self-supervised learning for dense depth estimation in monocular endoscopy. In *OR 2.0*, pages 128–138. Springer, 2018.

[150] J. E. Lovie-Kitchin, J. C. Mainstone, J. Robinson, and B. Brown. What Areas of of the visual Field are Important for Mobility in Low Vision Patients. *Clinical Vision Sciences*, 5(3):249–263, 1990.

[151] A. Lucero and A. Vetek. Notifeye: using interactive glasses to deal with notifications while walking in public. In *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology*, page 17. ACM, 2014.

[152] K. Luyten, D. Degraen, G. Rovelo Ruiz, S. Coppers, and D. Vanacken. Hidden in plain sight: an exploration of a visual language for near-eye out-of-focus displays in the peripheral view. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 487–497. ACM, 2016.

[153] MagicLeap. [Online], 2020. Availble: https://developer.magicleap.com/en-us/learn/guides/lumin-sdk-image-tracking (visited on 2020-03-19).

[154] P. Maier, A. Dey, C. A. Waechter, C. Sandor, M. Tönnis, and G. Klinker. An empiric evaluation of confirmation methods for optical see-through head-mounted display calibration. In *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*, pages 267–268. IEEE, 2011.

[155] J. Mamoun and M. E. Wilkinson. Technical aspects and clinical usage of keplerian and galilean binocular surgical loupe telescopes used in dentistry or medicine. 2013.

[156] H. B. Mann and D. R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pages 50–60, 1947.

[157] S. Martin. The role of the first assistant in robotic assisted surgery. *British Journal of Perioperative Nursing*, 14(4):159–163, 2004.

[158] A. Martin-Gonzalez, S.-M. Heining, and N. Navab. Head-mounted virtual loupe with sight-based activation for surgical applications. In *2009 8th IEEE international symposium on mixed and augmented reality*, pages 207–208. IEEE, 2009.

[159] A. Mason, R. Paulsen, J. M. Babuska, S. Rajpal, S. Burneikiene, E. L. Nelson, and A. T. Villavicencio. The accuracy of pedicle screw placement using intraop-

erative image guidance systems: A systematic review. *Journal of Neurosurgery: Spine*, 20(2):196–203, 2014.

[160] F. O. Matu, M. Thøgersen, B. Galsgaard, M. M. Jensen, and M. Kraus. Stereoscopic augmented reality system for supervised training on minimal invasive surgery robots. In *Virtual Reality International Conference*, page 33. ACM, 2014.

[161] G. Megali, V. Ferrari, C. Freschi, B. Morabito, F. Cavallo, G. Turini, E. Troia, C. Cappelli, A. Pietrabissa, O. Tonet, et al. EndoCAS navigator platform: a common platform for computer and robotic assistance in minimally invasive surgery. *IJCARS*, 4(3):242–251, 2008.

[162] C. Meng, T. Wang, W. Chou, S. Luan, Y. Zhang, and Z. Tian. Remote surgery case: robot-assisted teleneurosurgery. In *ICRA*, volume 1, pages 819–823. IEEE, 2004.

[163] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino. Augmented reality: A class of displays on the reality-virtuality continuum. In *Telemanipulator and Telepresence Technologies*, volume 2351, pages 282–292. International Society for Optics and Photonics, 1995.

[164] T. Miyaki and J. Rekimoto. LiDARMAN: Reprogramming Reality with Egocentric Laser Depth Scanning. In *ACM SIGGRAPH 2016 Emerging Technologies*, pages 15:1–15:2, 2016.

[165] O. Mohareri, J. Ischia, P. C. Black, C. Schneider, J. Lobo, L. Goldenberg, and S. E. Salcudean. Intraoperative registered transrectal ultrasound guidance for robot-assisted laparoscopic radical prostatectomy. *The Journal of Urology*, 193(1):302–312, 2015.

[166] O. Mohareri, G. Nir, J. Lobo, R. Savdie, P. Black, and S. Salcudean. A system for MR-ultrasound guidance during robot-assisted laparoscopic radical prostatectomy. In *MICCAI*, pages 497–504. Springer, 2015.

[167] O. Mohareri, C. Schneider, T. K. Adebar, M. C. Yip, P. Black, C. Y. Nguan, D. Bergman, J. Seroger, S. DiMaio, and S. E. Salcudean. Ultrasound-based image guidance for robot-assisted laparoscopic radical prostatectomy: initial in-vivo results. In *IPCAI*, pages 40–50. Springer, 2013.

[168] Mojo Vision Inc. [Online], 2020. Availble: https://www.mojo.vision/mojo-lens (visited on 2020-03-17).

[169] K. R. Moser. Quantification of error from system and environmental sources in Optical See-Through head mounted display calibration methods. In *Virtual Reality (VR), 2014 IEEE*, pages 137–138. IEEE, 2014.

[170] K. R. Moser. *Towards system agnostic calibration of optical see-through head-mounted displays for augmented reality.* PhD thesis, Mississippi State University, 2016.

[171] K. R. Moser, M. Axholt, and J. E. Swan. Baseline SPAAM calibration accuracy and precision in the absence of human postural sway error. In *Virtual Reality (VR), 2014 IEEE*, pages 99–100. IEEE, 2014.

[172] K. R. Moser and J. E. Swan. Evaluation of hand and stylus based calibration for optical see-through head-mounted displays using leap motion. In *Virtual Reality (VR), 2016 IEEE*, pages 233–234. IEEE, 2016.

[173] F. Mourgues, È. Coste-Manière, C. Team, et al. Flexible calibration of actuated stereoscopic endoscope for overlay in robot assisted surgery. In *MICCAI*, pages 25–34. Springer, 2002.

[174] F. Mourgues, F. Devemay, and E. Coste-Maniere. 3D reconstruction of the operating field for image overlay in 3D-endoscopic surgery. In *International Symposium on Augmented Reality*, pages 191–192. IEEE, 2001.

[175] National Health Expenditure Projections 2018-2027. [Online], 2017. Availble: https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/NationalHealthExpendData/Downloads/ForecastSummary.pdf (visited on 2020-01-27).

[176] N. Navab, T. Blum, L. Wang, A. Okur, and T. Wendler. First deployments of augmented reality in operating rooms. *Computer*, 45(7):48–55, 2012.

[177] N. Navab, S. Zokai, Y. Genc, and E. M. Coelho. An On-line Evaluation System for Optical See-through Augmented Reality. In *Proceedings of the IEEE Virtual Reality 2004*, VR '04, pages 245–, Washington, DC, USA, 2004. IEEE Computer Society.

[178] R. Nayyar, S. Yadav, P. Singh, and P. N. Dogra. Impact of assistant surgeon on outcomes in robotic surgery. *Indian Journal of Urology (IJU)*, 32(3):204, 2016.

[179] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *ISMAR*, pages 127–136. IEEE, 2011.

[180] J. Newman, M. Wagner, M. Bauer, A. MacWilliams, T. Pintaric, D. Beyer, D. Pustka, F. Strasser, D. Schmalstieg, and G. Klinker. Ubiquitous tracking for augmented reality. In *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on*, pages 192–201. IEEE, 2004.

[181] S. Niederhauser and D. S. Mojon. Normal Isopter Position in the Peripheral Visual Field in Goldmann Kinetic Perimetry. *Ophthalmologica*, 216(6):406–408, 2002.

[182] E. Niforatos, A. Fedosov, I. Elhart, and M. Langheinrich. Augmenting skiers' peripheral perception. In *Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pages 114–121. ACM, 2017.

[183] M. S. Nosrati, R. Abugharbieh, J.-M. Peyrat, J. Abinahed, O. Al-Alao, A. Al-Ansari, and G. Hamarneh. Simultaneous multi-structure segmentation and 3D nonrigid pose estimation in image-guided robotic surgery. *TMI*, 35(1):1–12, 2016.

[184] M. S. Nosrati, A. Amir-Khalili, J.-M. Peyrat, J. Abinahed, O. Al-Alao, A. Al-Ansari, R. Abugharbieh, and G. Hamarneh. Endoscopic scene labelling and augmentation using intraoperative pulsatile motion and colour appearance cues with preoperative anatomical priors. *IJCARS*, 11(8):1409–1418, 2016.

[185] K. N. Ogle. Disparity limits of stereopsis. *AMA archives of ophthalmology*, 48(1):50–60, 1952.

[186] A. M. Okamura. Haptic feedback in robot-assisted minimally invasive surgery. *Current Opinion in Urology*, 19(1):102, 2009.

[187] D. Ong, N. H. Chua, and K. Vissers. Percutaneous disc decompression for lumbar radicular pain: a review article. *Pain Practice*, 16(1):111–126, 2016.

[188] J. Orlosky, Q. Wu, K. Kiyokawa, H. Takemura, and C. Nitschke. Fisheye Vision: Peripheral Spatial Compression for Improved Field of View in Head Mounted Displays. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pages 54–61, 2014.

[189] C. B. Owen, J. Zhou, A. Tang, and F. Xiao. Display-relative calibration for

optical see-through head-mounted displays. In *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on*, pages 70–78. IEEE, 2004.

[190] H. R. Patel, A. Linares, and J. V. Joseph. Robotic and laparoscopic surgery: cost and training. *Surgical Oncology*, 18(3):242–246, 2009.

[191] V. Penza, E. De Momi, N. Enayati, T. Chupin, J. Ortiz, and L. S. Mattos. EnViSoRS: enhanced vision system for robotic surgery. a user-defined safety volume tracking to minimize the risk of intraoperative bleeding. *Frontiers in Robotics and AI*, 4:15, 2017.

[192] V. Penza, J. Ortiz, L. S. Mattos, A. Forgione, and E. De Momi. Dense soft tissue 3d reconstruction refined with super-pixel segmentation for robotic abdominal surgery. *International journal of computer assisted radiology and surgery*, 11(2):197–206, 2016.

[193] P. Perrin, M. Eichenberger, K. Neuhaus, and A. Lussi. Visual acuity and magnification devices in dentistry. *Swiss Dent J*, 126(3):222–35, 2016.

[194] M. Peterson and Z. Basrai. Introduction to bedside ultrasound. *Clinical Emergency Radiology*, 195, 2017.

[195] B. Peuchot, A. Tanguy, and M. Eude. Virtual reality as an operative tool during

scoliosis surgery. In *International Conference on Computer Vision, Virtual Reality, and Robotics in Medicine*, pages 549–554. Springer, 1995.

[196] A. Plopski, C. Nitschke, K. Kiyokawa, D. Schmalstieg, and H. Takemura. Hybrid Eye Tracking: Combining Iris Contour and Corneal Imaging. In *Proceedings of the International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments*, pages 183–190, October 2015.

[197] Pokemon Go. [Online], 2016. Available: https://www.pokemongo.com/en-us/ (visited on 2020-01-27).

[198] B. Poppinga, N. Henze, J. Fortmann, W. Heuten, and S. Boll. Ambiglasses-information in the periphery of the visual field. In *Mensch & Computer*, pages 153–162, 2012.

[199] F. Porpiglia, E. Checcucci, D. Amparore, R. Autorino, A. Piana, A. Bellin, P. Piazzolla, F. Massa, E. Bollito, D. Gned, et al. Augmented-reality robot-assisted radical prostatectomy using hyper-accuracy three-dimensional reconstruction (HA 3D^{TM}) technology: a radiological and pathological study. *BJU International*, 2018.

[200] F. Porpiglia, E. Checcucci, D. Amparore, M. Manfredi, F. Massa, P. Piazzolla, D. Manfrin, A. Piana, D. Tota, E. Bollito, et al. Three-dimensional elastic augmented-reality robot-assisted radical prostatectomy using hyperaccuracy

three-dimensional reconstruction technology: A step further in the identification of capsular involvement. *European Urology*, 2019.

[201] F. Porpiglia, C. Fiori, E. Checcucci, D. Amparore, and R. Bertolo. Hyperaccuracy three-dimensional reconstruction is able to maximize the efficacy of selective clamping during robot-assisted partial nephrectomy for complex renal masses. *European Urology*, 2018.

[202] P. Pratt, E. Mayer, J. Vale, D. Cohen, E. Edwards, A. Darzi, and G.-Z. Yang. An effective visualisation and registration system for image-guided robotic partial nephrectomy. *Journal of Robotic Surgery*, 6(1):23–31, 2012.

[203] G. A. Puerto-Souza and G. L. Mariottini. Toward long-term and accurate augmented-reality display for minimally-invasive surgery. In *ICRA*, pages 5384–5389. IEEE, 2013.

[204] D. Putzer, S. Klug, J. Moctezuma, M. Nogler, and I. Stryker. The use of time of flight camera for soft tissue tracking during minimally invasive hip arthroplasty. *Roboter-Assistenten werden sensitiv.*, page 130, 2013.

[205] D. Putzer, S. Klug, J. L. Moctezuma, and M. Nogler. The use of time-of-flight camera for navigating robots in computer-aided surgery: Monitoring the soft tissue envelope of minimally invasive hip approach in a cadaver study. *Surgical Innovation*, 21(6):630–636, 2014.

[206] L. Qian, E. Azimi, P. Kazanzides, and N. Navab. Comprehensive tracker based display calibration for holographic optical see-through head-mounted display. *arXiv preprint arXiv:1703.05834*, 2017.

[207] L. Qian, A. Deguet, and P. Kazanzides. ARssist: augmented reality on a head-mounted display for the first assistant in robotic surgery. *Healthcare Technology Letters*, 5(5):194–200, 2018.

[208] L. Qian, A. Winkler, B. Fuerst, P. Kazanzides, and N. Navab. Modeling physical structure as additional constraints for stereoscopic optical see-through head-mounted display calibration. In *Mixed and Augmented Reality (ISMAR-Adjunct), 2016 IEEE International Symposium on*, pages 154–155. IEEE, 2016.

[209] L. Qian, A. Winkler, B. Fuerst, P. Kazanzides, and N. Navab. Reduction of interaction space in single point active alignment method for optical see-through head-mounted display calibration. In *Mixed and Augmented Reality (ISMAR-Adjunct), 2016 IEEE International Symposium on*, pages 156–157. IEEE, 2016.

[210] L. Qian, J. Y. Wu, S. DiMaio, N. Navab, and P. Kazanzides. A review of augmented reality in robotic-assisted surgery. *IEEE Transactions on Medical Robotics and Bionics*, 2019.

[211] L. Qian, X. Zhang, A. Deguet, and P. Kazanzides. ARAMIS: Augmented Reality Assistance for Minimally Invasive Surgery Using a Head-Mounted Display. In *MICCAI*, pages 74–82. Springer, 2019.

[212] I. Rakkolainen, R. Raisamo, M. Turk, and T. Höllerer. Field-of-view Extension for VR Viewers. In *Proceedings of the International Academic Mindtrek Conference*, pages 227–230, 2017.

[213] I. Rakkolainen, M. Turk, and T. Höllerer. A Superwide-FOV Optical Design for Head-Mounted Displays. In *Proceedings of the International Conference on Artificial Reality and Telexistence and the Eurographics Symposium on Virtual Environments*, pages 45–48, 2016.

[214] R. Raskar, G. Welch, and H. Fuchs. Spatially augmented reality. In *IWAR*, pages 11–20, 1998.

[215] S. Reed, O. Kreylos, S. Hsi, L. Kellogg, G. Schladow, M. Yikilmaz, H. Segale, J. Silverman, S. Yalowitz, and E. Sato. Shaping watersheds exhibit: An interactive, augmented reality sandbox for advancing earth science education. In *AGU Fall Meeting Abstracts*, 2014.

[216] P. Renner and T. Pfeiffer. Attention guiding techniques using peripheral vision and eye tracking for feedback in augmented-reality-based assistance systems. In *3D User Interfaces (3DUI), 2017 IEEE Symposium on*, pages 186–194. IEEE, 2017.

[217] J. B. Roerdink and A. Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta informaticae*, 41(1, 2):187–228, 2000.

[218] J. P. Rolland and H. Fuchs. Optical versus video see-through head-mounted displays in medical visualization. *Presence: Teleoperators & Virtual Environments*, 9(3):287–309, 2000.

[219] E. Rosen. The invention of eyeglasses. *Journal of the history of medicine and allied sciences*, 11(1):13–46, 1956.

[220] R. Rosenholtz. Capabilities and Limitations of Peripheral Vision. *Annual Review of Vision Science*, 2(1):437–457, 2016.

[221] R. Rosenholtz, Y. Li, and L. Nakano. Measuring visual clutter. *Journal of Vision*, 7(2):17–17, 2007.

[222] C. M. Rumack, S. R. Wilson, and J. W. Charboneau. *Diagnostic ultrasound*. Elsevier Mosby, 2005.

[223] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison. Slam++: Simultaneous Localisation and Mapping at the Level of Objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1352–1359, 2013.

[224] G. Samei, O. Goksel, J. Lobo, O. Mohareri, P. Black, R. Rohling, and S. Salcudean. Real-time FEM-based registration of 3D to 2.5 D transrectal ultrasound images. *TMI*, 2018.

[225] T. M. Schmidt, S.-K. Chen, and S. Hattar. Intrinsically Photosensitive Retinal

Ganglion Cells: Many Subtypes, Diverse Functions. *Trends in Neurosciences*, 34(11):572–580, 2011.

[226] C. M. Schneider, P. D. Peng, R. H. Taylor, G. W. Dachs II, C. J. Hasser, S. P. DiMaio, and M. A. Choti. Robot-assisted laparoscopic ultrasonography for hepatic surgery. *Surgery*, 151(5):756–762, 2012.

[227] A. Sengiku, M. Koeda, A. Sawada, J. Kono, N. Terada, T. Yamasaki, K. Mizushino, T. Kunii, K. Onishi, H. Noborio, et al. Augmented reality navigation system for robot-assisted laparoscopic partial nephrectomy. In *International Conference of Design, User Experience, and Usability*, pages 575–584. Springer, 2017.

[228] Sensorama. [Online]. Available: https://en.wikipedia.org/wiki/Sensorama (visited on 2020-01-27).

[229] O. Sgarbura and C. Vasilescu. The decisive role of the patient-side surgeon in robotic surgery. *Surgical Endoscopy*, 24(12):3149–3155, 2010.

[230] S. S. Shapiro and M. B. Wilk. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611, 1965.

[231] J. H. Shuhaiber. Augmented reality in surgery. *Archives of surgery*, 139(2):170–174, 2004.

[232] M. Simoes and C. G. Cao. Leonardo: a first step towards an interactive decision aid for port-placement in robotic surgery. In *International Conference on Systems, Man, and Cybernetics*, pages 491–496. IEEE, 2013.

[233] R. Singla, P. Edgcumbe, P. Pratt, C. Nguan, and R. Rohling. Intra-operative ultrasound-based augmented reality guidance for laparoscopic surgery. *Healthcare Technology Letters*, 4(5):204–209, 2017.

[234] A. Sinha et al. *Deformable registration using shape statistics with applications in sinus surgery*. PhD thesis, Johns Hopkins University, 2018.

[235] K. Someya, Y. Hiroi, M. Yamada, and Y. Itoh. Ostnet: Calibration method for optical see-through head-mounted displays via non-parametric distortion map generation. In *2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pages 259–260. IEEE, 2019.

[236] J. Song, J. Wang, L. Zhao, S. Huang, and G. Dissanayake. MIS-SLAM: Real-time large-scale dense deformable SLAM system in minimal invasive surgery based on heterogeneous computing. *RAL*, 3(4):4068–4075, 2018.

[237] M. Sonka, V. Hlavac, and R. Boyle. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[238] R. H. Spector. Visual Fields. In H. K. Walker, W. D. Hall, and J. W. Hurst,

editors, *Clinical Methods: The History, Physical, and Laboratory Examinations*, chapter 116, pages 565–572. Boston: Butterworths, 1990.

[239] S. Speidel, S. Krappe, S. Röhl, S. Bodenstedt, B. Müller-Stich, and R. Dillmann. Robust feature tracking for endoscopic pose estimation and structure recovery. In *Medical Imaging 2013: Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 8671, page 867102. International Society for Optics and Photonics, 2013.

[240] C. Staub, C. Lenz, G. Panin, A. Knoll, and R. Bauernschmitt. Contour-based surgical instrument tracking supported by kinematic prediction. In *2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 746–752. IEEE, 2010.

[241] M. Stevenson, T. Gomersall, M. L. Jones, A. Rawdin, M. Hernández, S. Dias, D. Wilson, and A. Rees. Percutaneous vertebroplasty and percutaneous balloon kyphoplasty for the treatment of osteoporotic vertebral fractures: a systematic review and cost-effectiveness analysis. In *Percutaneous vertebroplasty and percutaneous balloon kyphoplasty for the treatment of osteoporotic vertebral fractures: a systematic review and cost-effectiveness analysis*. NIHR Journals Library, 2014.

[242] R. Stoakley, M. J. Conway, and R. Pausch. Virtual Reality on a WIM: Interac-

tive Worlds in Miniature. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 265–272, 1995.

[243] D. Stoyanov and G.-Z. Yang. Stabilization of image motion for robotic assisted beating heart surgery. In *MICCAI*, pages 417–424. Springer, 2007.

[244] H. Strasburger, I. Rentschler, and M. Jüttner. Peripheral Vision and Pattern Recognition: A Review. *Journal of vision*, 11(5):13:1–13:82, 2011.

[245] F. F. Strobl, S. M. Haeussler, P. M. Paprottka, R.-T. Hoffmann, O. Pieske, M. F. Reiser, and C. G. Trumm. Technical and clinical outcome of percutaneous ct fluoroscopy-guided screw placement in unstable injuries of the posterior pelvic ring. *Skeletal radiology*, 43(8):1093–1100, 2014.

[246] L.-M. Su, B. P. Vagvolgyi, R. Agarwal, C. E. Reiley, R. H. Taylor, and G. D. Hager. Augmented reality during robot-assisted laparoscopic partial nephrectomy: toward real-time 3D-CT to stereoscopic video registration. *Urology*, 73(4):896–900, 2009.

[247] M. Sukan, C. Elvezio, O. Oda, S. Feiner, and B. Tversky. Parafrustum: Visualization Techniques for Guiding a User to a Constrained Set of Viewing Positions and Orientations. In *Proceedings of the Annual ACM symposium on User Interface Software and Technology*, pages 331–340, 2014.

[248] I. E. Sutherland. A head-mounted three dimensional display. In *Fall Joint Computer Conference, part I*, pages 757–764. ACM, 1968.

[249] N. Suzuki, A. Hattori, K. Tanoue, S. Ieiri, K. Konishi, M. Tomikawa, H. Kenmotsu, and M. Hashizume. Scorpion shaped endoscopic surgical robot for NOTES and SPS with augmented reality functions. In *International Workshop on Medical Imaging and Virtual Reality*, pages 541–550. Springer, 2010.

[250] J. P. Szlyk, C. L. Mahler, W. Seiple, D. P. Edward, and J. T. Wilensky. Driving Performance of Glaucoma Patients Correlates with Peripheral Visual Field Loss. *Journal of glaucoma*, 14(2):145–150, 2005.

[251] V. Tacher, M. Lin, P. Desgranges, J.-F. Deux, T. Grünhagen, J.-P. Becquemin, A. Luciani, A. Rahmouni, and H. Kobeiter. Image guidance for endovascular repair of complex aortic aneurysms: comparison of two-dimensional and three-dimensional angiography and image fusion. *Journal of Vascular and Interventional Radiology*, 24(11):1698–1706, 2013.

[252] A. Tang, J. Zhou, and C. Owen. Evaluation of calibration procedures for optical see-through head-mounted displays. In *IEEE/ACM Intl. Symp. on Mixed and Augmented Reality (ISMAR)*, page 161, 2003.

[253] R. H. Taylor. Computer-integrated interventional medicine: A 30 year perspective. In *Handbook of Medical Image Computing and Computer Assisted Intervention*, pages 599–624. Elsevier, 2020.

[254] R. H. Taylor, S. Lavealle, G. C. Burdea, and R. Mosges. *Computer-integrated surgery: technology and clinical applications*. Mit Press, 1995.

[255] R. H. Taylor, A. Menciassi, G. Fichtinger, P. Fiorini, and P. Dario. Medical robotics and computer-integrated surgery. In *Handbook of Robotics*, pages 1657–1684. Springer, 2016.

[256] M. Tonnis and G. Klinker. Effective Control of a Car Driver's Attention for Visual and Acoustic Guidance Towards the Direction of Imminent Dangers. In *Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 13–22, 2006.

[257] M. Trapp, L. Schneider, C. Lehmann, N. Holz, and J. Döllner. Strategies for Visualising 3D Points-of-Interest on Mobile Devices. *Journal of Location Based Services*, 5(2):79–99, 2011.

[258] M. Tuceryan, Y. Genc, and N. Navab. Single-point active alignment method (spaam) for optical see-through hmd calibration for augmented reality. *Presence: Teleoperators and Virtual Environments*, 11(3):259–276, 2002.

[259] G. Turchetti, I. Palla, F. Pierotti, and A. Cuschieri. Economic evaluation of da Vinci-assisted robotic surgery: a systematic review. *Surgical Endoscopy*, 26(3):598–606, 2012.

[260] S. Umeyama. Least-squares estimation of transformation parameters between

two point patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 13(4):376–380, 1991.

[261] I. Urlic, Ž. Verzak, and D. N. Vranic. Measuring the influence of galilean loupe system on near visual acuity of dentists under simulated clinical conditions. *Acta stomatologica Croatica*, 50(3):235, 2016.

[262] M. N. van Oosterom, M. A. Engelen, N. S. van den Berg, G. H. KleinJan, H. G. van der Poel, T. Wendler, C. J. H. van de Velde, N. Navab, and F. W. B. van Leeuwen. Navigation of a robot-integrated fluorescence laparoscope in preoperative SPECT/CT and intraoperative freehand SPECT imaging data: a phantom study. *Journal of Biomedical Optics*, 21(8):086008, 2016.

[263] F. Vargas-Martin, E. Peli, et al. Augmented-View for Restricted Visual Field: Multiple Device Implementations. *Optometry and Vision Science*, 79(11):715–723, 2002.

[264] L. Vincent. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *IEEE transactions on image processing*, 2(2):176–201, 1993.

[265] F. Volonté, N. C. Buchs, F. Pugin, J. Spaltenstein, M. Jung, O. Ratib, and P. Morel. Stereoscopic augmented reality for da Vinci^{TM} robotic biliary surgery. *International Journal of Surgery Case Reports*, 4(4):365–367, 2013.

[266] F. Volonté, N. C. Buchs, F. Pugin, J. Spaltenstein, B. Schiltz, M. Jung, M. Hagen, O. Ratib, and P. Morel. Augmented reality to the rescue of the minimally invasive surgeon. the usefulness of the interposition of stereoscopic images in the Da Vinci^{TM} robotic console. *IJMRCAS*, 9(3):e34–e38, 2013.

[267] F. Volonté, F. Pugin, P. Bucher, M. Sugimoto, O. Ratib, and P. Morel. Augmented reality and image overlay navigation with OsiriX in laparoscopic and robotic surgery: not only a matter of fashion. *Journal of Hepato-biliary-pancreatic Sciences*, 18(4):506–509, 2011.

[268] F. Volonté, F. Pugin, N. C. Buchs, J. Spaltenstein, M. Hagen, O. Ratib, and P. Morel. Console-integrated stereoscopic Osirix 3D volume-rendered images for da Vinci colorectal robotic surgery. *Surgical Innovation*, 20(2):158–163, 2013.

[269] W. Vorraber, S. Voessner, G. Stark, D. Neubacher, S. DeMello, and A. Bair. Medical applications of near-eye display devices: an exploratory study. *International Journal of Surgery*, 12(12):1266–1272, 2014.

[270] A. Voruganti, R. Mayoral, S. Jacobs, R. Grunert, H. Moeckel, and W. Korb. Surgical cartographic navigation system for endoscopic bypass grafting. In *EMBS*, pages 1467–1470. IEEE, 2007.

[271] D. Wang, F. Bello, and A. Darzi. Augmented reality provision in robotically assisted minimally invasive surgery. In *International Congress Series*, volume 1268, pages 527–532. Elsevier, 2004.

[272] D. Wang, A. Faraci, F. Bello, and A. Darzi. Simulating tele-manipulator controlled tool-tissue interactions using a nonlinear FEM deformable model. *Studies in Health Technology and Informatics*, 119:565–567, 2006.

[273] H. Wang, F. Wang, A. P. Y. Leong, L. Xu, X. Chen, and Q. Wang. Precision insertion of percutaneous sacroiliac screws using a novel augmented reality-based navigation system: a pilot study. *International orthopaedics*, 40(9):1941–1947, 2016.

[274] J. Wang, L. Qian, E. Azimi, and P. Kazanzides. Prioritization and static error compensation for multi-camera collaborative tracking in augmented reality. In *Virtual Reality (VR)*. IEEE, 2017.

[275] Y.-Y. Wang, A. Kumar, K.-C. Liu, S.-W. Huang, C.-C. Huang, W.-C. Su, F.-L. Hsiao, and W.-N. Lie. Stereoscopic augmented reality for single camera endoscopy: a virtual study. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 6(2):182–191, 2018.

[276] D. Wardlaw, S. R. Cummings, J. Van Meirhaeghe, L. Bastian, J. B. Tillman, J. Ranstam, R. Eastell, P. Shabe, K. Talmadge, and S. Boonen. Efficacy and safety of balloon kyphoplasty compared with non-surgical care for vertebral compression fracture (free): a randomised controlled trial. *The Lancet*, 373(9668):1016–1024, 2009.

[277] O. Weede, M. Mehrwald, and H. Wörn. Knowledge-based system for port

placement and robot setup optimization in minimally invasive surgery. *IFAC*, 45(22):722–728, 2012.

[278] O. Weede, J. Wünscher, H. Kenngott, B. Müller-Stich, and H. Wörn. Knowledge-based planning of port positions for minimally invasive surgery. In *Conference on Cybernetics and Intelligent Systems*, pages 12–17. IEEE, 2013.

[279] C. Wei and A. Y. Wu. Surgical loupe usage among oculoplastic surgeons in north america. *Canadian Journal of Ophthalmology*, 53(2):139–144, 2018.

[280] R. Wen, C.-B. Chng, and C.-K. Chui. Augmented reality guidance with multimodality imaging data and depth-perceived interaction for robot-assisted surgery. *Robotics*, 6(2):13, 2017.

[281] R. Wen, C.-B. Chng, C.-K. Chui, K.-B. Lim, S.-H. Ong, and S.-Y. Chang. Robot-assisted RF ablation with interactive planning and mixed reality guidance. In *International Symposium on System Integration*, pages 31–36. IEEE, 2012.

[282] R. Wen, C.-K. Chui, S.-H. Ong, K.-B. Lim, and S. K.-Y. Chang. Projection-based visual guidance for robot-aided RF needle insertion. *IJCARS*, 8(6):1015–1025, 2013.

[283] R. Wen, B. P. Nguyen, C.-B. Chng, and C.-K. Chui. In situ spatial AR surgical

planning using projector-kinect system. In *Symposium on Information and Communication Technology*, pages 164–171. ACM, 2013.

[284] R. Wen, W.-L. Tay, B. P. Nguyen, C.-B. Chng, and C.-K. Chui. Hand gesture guided robot-assisted surgery based on a direct augmented reality interface. *Computer Methods and Programs in Biomedicine*, 116(2):68–80, 2014.

[285] R. Wen, L. Yang, C.-K. Chui, K.-B. Lim, and S. Chang. Intraoperative visual guidance and control interface for augmented reality robotic surgery. In *International Conference on Control and Automation*, pages 947–952. IEEE, 2010.

[286] B. Wentink. Eye-hand coordination in laparoscopy-an overview of experiments and supporting aids. *Minimally Invasive Therapy & Allied Technologies*, 10(3):155–162, 2001.

[287] J. D. Westwood et al. The mini-screen: an innovative device for computer assisted surgery systems. *Medicine Meets Virtual Reality 13: The Magical Next Becomes the Medical Now*, 111:314, 2005.

[288] F. Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer, 1992.

[289] F. Wilcoxon, S. Katti, and R. A. Wilcox. Critical values and probability levels

for the wilcoxon rank sum test and the wilcoxon signed rank test. *Selected tables in mathematical statistics*, 1:171–259, 1970.

[290] G. R. Wilensky. Robotic surgery: an example of when newer is not always better but clearly more expensive. *The Milbank Quarterly*, 94(1):43, 2016.

[291] R. L. Woods, I. Fetchenheuer, F. Vargas-Martín, and E. Peli. The Impact of Non-Immersive Head-Mounted Displays (HMDs) on the Visual Field. *Journal of the Society for Information Display*, 11(1):191–198, 2003.

[292] H. Wörn and J. Mühling. Computer- and robot-based operation theatre of the future in cranio-facial surgery. In *International Congress Series*, volume 1230, pages 753–759. Elsevier, 2001.

[293] H. Wörn and O. Weede. Optimizing the setup configuration for manual and robotic assisted minimally invasive surgery. In *World Congress on Medical Physics and Biomedical Engineering*, pages 55–58. Springer, 2009.

[294] J. Y. Wu, A. Tuomi, M. D. Beland, J. Konrad, D. Glidden, D. Grand, and D. Merck. Quantitative analysis of ultrasound images for computer-aided diagnosis. *JMI*, 3(1):014501, 2016.

[295] R. Xiao and H. Benko. Augmenting the field-of-view of head-mounted displays with sparse peripheral displays. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1221–1232. ACM, 2016.

[296] T. Yamamoto, N. Abolhassani, S. Jung, A. M. Okamura, and T. N. Judkins. Augmented reality and haptic interfaces for robot-assisted surgery. *IJMRCAS*, 8(1):45–56, 2012.

[297] T. Yamamoto, B. Vagvolgyi, K. Balaji, L. L. Whitcomb, and A. M. Okamura. Tissue property estimation and graphical display for teleoperated robot-assisted surgery. In *ICRA*, pages 4239–4245. IEEE, 2009.

[298] Z. Yaniv and K. Cleary. Image-guided procedures: A review. *Computer Aided Interventions and Medical Robotics*, 3(1-63):7, 2006.

[299] Y. Yano, J. Orlosky, K. Kiyokawa, and H. Takemura. Dynamic View Expansion for Improving Visual Search in Video See-Through AR. In *Proceedings of the International Conference on Artificial Reality and Telexistence and the Eurographics Symposium on Virtual Environments*, pages 57–60, December 2016.

[300] Y. Yimin, R. Zhiwei, M. Wei, and R. Jha. Current status of percutaneous vertebroplasty and percutaneous kyphoplasty–a review. *Medical science monitor: international medical journal of experimental and clinical research*, 19:826, 2013.

[301] J. W. Yoon, R. E. Chen, P. K. Han, P. Si, W. D. Freeman, and S. M. Pirris. Technical feasibility and safety of an intraoperative head-up display device during spine instrumentation. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 13(3):e1770, 2017.

[302] B. Zeng, F. Meng, H. Ding, and G. Wang. A surgical robot with augmented reality visualization for stereoelectroencephalography electrode implantation. *IJCARS*, 12(8):1355–1368, 2017.

[303] N. Zevallos, R. A. Srivatsan, H. Salman, L. Li, J. Qian, S. Saxena, M. Xu, K. Patath, and H. Choset. A surgical system for automatic registration, stiffness mapping and dynamic image overlay. In *ISMR*, pages 1–6. IEEE, 2018.

[304] L. Zhang, M. Ye, P. Giataganas, M. Hughes, A. Bradu, A. Podoleanu, and G.-Z. Yang. From macro to micro: Autonomous multiscale image fusion for robotic surgery. *RAM*, 24(2):63–72, 2017.

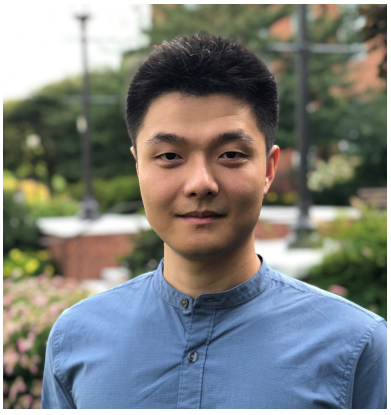
[305] L. Zhang, M. Ye, P. Giataganas, M. Hughes, and G.-Z. Yang. Autonomous scanning for endomicroscopic mosaicing and 3D fusion. In *ICRA*, pages 3587–3593. IEEE, 2017.

[306] Z. Zhang. Microsoft kinect sensor and its effect. *Multimedia*, 19(2):4–10, 2012.

[307] C. Zhou, M. Zhu, Y. Shi, L. Lin, G. Chai, Y. Zhang, and L. Xie. Robot-assisted surgery for mandibular angle split osteotomy using augmented reality: Preliminary results on clinical animal experiment. *Aesthetic Plastic Surgery*, 41(5):1228–1236, 2017.

[308] E. Zhu, A. Hadadgar, I. Masiello, and N. Zary. Augmented reality in healthcare education: an integrative review. *PeerJ*, 2:e469, 2014.

Vita



Long Qian received the B.S. degree in the Department of Electronics Engineering from Tsinghua University in 2015, and enrolled in the Computer Science Ph.D. program at the Johns Hopkins University in the same year. He joined the Laboratory for Computational Sensing and Robotics (LCSR) to work on augmented reality, especially with head-mounted displays and the clinical applications, with the supervision by Prof. Peter Kazanzides and Prof. Nassir Navab.